T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

TASARSIZ AĞLARIN GÜVENLİĞİ

Mohammad ABO SAALEEK

Bilgisayar Mühendisliği Anabilim Dalı

Danışman

Prof.Dr. Abdül Halim ZAİM

Temmuz, 2014

İSTANBUL

Bu çalışma 08/07/2014 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği programında Doktora Tezi olarak kabul edilmiştir.

Tez Jürisi:

Prof. Dr. A.Halim ZAİM (Danışman)
İstanbul Ticaret Üniversitesi
Mühendislik ve Tasarım Fakültesi

Prof. Dr. Ahmet SERTBAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof. Dr. Gökhan UZGÖREN
Gedik Üniversitesi
Mühendislik Fakültesi

Prof.Dr. Selim AKYOKUŞ
Doğuş Üniversitesi
Mühendislik Fakültesi

Yrd. Doç. Dr. Muhammed Ali AYDIN
İstanbul Üniversitesi
Mühendislik Fakültesi

ÖNSÖZ

Yüksek lisans ve Doktora öğrenimim sırasında ve tez çalışmalarım boyunca her türlü maddi manevi destek ve yardımlarını esirgemeyen , ayrıca tecrübelerinden yararlanırken göstermiş olduğu hoşgörü ve sabrından dolayı, çok değerli hocam **Prof.Dr.Abdül Halim ZAİM**'e en içten dileklerimle teşekkür ederim.

Bu çalışma boyunca yardımlarını esirgemeyen çalışma arkadaşlarıma ve çalışmamın uygulama kısmını destekleyen İstanbul Üniversitesi'ne teşekkürü borç bilirim.

Bu günlere gelmemde büyük pay sahibi olan aileme ve dostlarıma teşekkürlerimi sunarım.

Temmuz, 2014 **Mohammad ABO SAALEEK**

İÇİNDEKİLER

Sayfa No

ŞEKİL LİSTESİ

Sayfa No

TABLO LİSTESİ

Sayfa No

SİMGE VE KISALTMA LİSTESİ

Kısaltmalar **Açıklama**

AODV **:** Ad hoc On Demand Distance Vector routing protocol
DSDV **:** Destination-Sequenced Distance-Vector
DSR **:** Dynamic Source Routing
FAODV : Flood Ad hoc On Demand Distance Vector routing protocol
FLSL **:** Fuzzy logic based security-level routing protocol
RADIUS **:**Remote Authentication Dial In User Service
LAN **:**Local Area Network
SAR **:** Security Aware Routing
SAODV **:** Secure Ad hoc On Demand Distance Vector routing protocol
SEAD **:** Secure Efficient Distance Vector Routing
SLSP **:** Secure link state protocol
SRP **:** The Secure Routing Protocol

ÖZET

DOKTORA TEZİ

TASARSIZ AĞLARIN GÜVENLİĞİ

Mohammad ABO SAALEEK

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Prof.Dr. Abdül Halim ZAİM

Tasarsız ağların geleneksel kablosuz ağlarla karşılaştırıldığında bazı avantajları bulunmaktadır. Bunlar, kolay gelişim, hızlı gelişim, sabit altyapı üzerinde azaltılmış bağımlılığı içerir. Tasarsız ağlarda yönlendirme protokollerinin kalite ve etkinliğini geliştirmek için birçok çalışma yapılmıştır. Ancak, açık eşler arası yapı, dinamik ağ topolojisi, ortak kablosuz ortam ve sınırlı kaynak (pil, hafıza ve hesaplama gücü) gibi Tasarsız ağ topolojilerinin benzersiz özellikleri, güvenlik tasarımı için kolay olmayan birçok zorluk oluşturur.

Bu zorluk ve özellikler, tasarsız ağların geniş çaplı koruma sağlamak ve istenilen ağ performansına sahip olmasını gerektirir. Bu tezde Tasarsız Talep Esaslı Uzaklık Vektörü – AODV, Güvenli Tasarsız Talep Üzerine Uzaklık Vektör yönlendirme protokolü –SAODV, Güvenli Etkili Uzaklık Vektör Yönlendirme – SEAD, Dinamik Güvenli Yönlendirme – DSR, Varış Yeri-Sıralı Uzaklık-Vektörü – çok sık kullanılan ağ simülatörü NS2 ile gerçeklenmiş DSDV gibi Tasarsız ağlardaki uygun güvenlik yönlendirme protokolleri incelenmiştir.Performans ve güvenlik ile ilgili hususları çalışabılmek amaçıyla, güvenli yol bulma protokolünün senaryoya dayalı similasyon ile analizi yapılmış ve sonuçlar Flood Ad Hoc On Demand Distance Vector routing protocol (FAODV) gibi geleneksel güvenli olmayan yol bulma protokolleri, ile karşılaştırılmıştır. gerçekleşmiş

Denemeler için kullanılan senaryolar, çelişen ihtiyaçları olma eğiliminde olan savaş alanı ve kurtarma operasyonları gibi kritik gerçek hayat uygulamalarını betimlemektedir. Performans ve güvenlik arasındaki kıyaslamayı dayalı analizle, incelenmekte olan yönlendirme protokollerinin uygulanabilirliğine ilişkin bilgi vermek için gerçekleştirilmiştir.

Temmuz 2014, 118 Sayfa.

Anahtar kelimeler: AODV , FAODV , AD HOC

SUMMARY

Ph.D. THESIS

ADHOC NETWORK SECURITY

Mohammad ABO SAALEEK

İstanbul University

Institute of Graduate Studies in Science and Engineering

Department of computer Engineering

Supervisor : Prof.Dr. Abdül Halim ZAİM

Ad hoc networks have several advantage compared with the traditional wireless networks . These include ease of development , speed of development and decreased dependency on fixed infrastructure. There have been many studies done in this area to improve the quality and the efficiency of the routing protocols in Ad hoc networks. However unique the characteristics of Ad hoc networks topology such as open peer-to-peer architecture, dynamic network topology, shared wireless medium and limited resource (battery, memory, and computation power) pose a number of non-trivial challenges to security design.

These challenges and characteristics require Ad hoc networks to provide broad protection and desirable network performance. In this thesis we examine the available secure routing protocols in Ad hoc networks such as Ad hoc On Demand Distance Vector routing protocol – AODV, Secure Ad hoc On Demand Distance Vector routing protocol – SAODV, Secure Efficient Distance Vector Routing – SEAD , *Dynamic Source Routing* – DSR, *Destination-Sequenced Distance-Vector* – DSDV implemented by a widely-used network simulator NS2. Study performance and security issues, a scenario based in order to simulation analysis of a secure routing protocol is performed and the results are compared with traditional non-secure routing protocols to produce Flood Ad hoc On Demand Distance Vector routing protocol – FAODV.

The scenarios used for the experiments depict critical real-world applications such as battlefield and rescue operations, which tend to have contradicting needs. An analysis of the tradeoffs between performance and security is done to gain an insight into the applicability of the routing protocols under consideration.

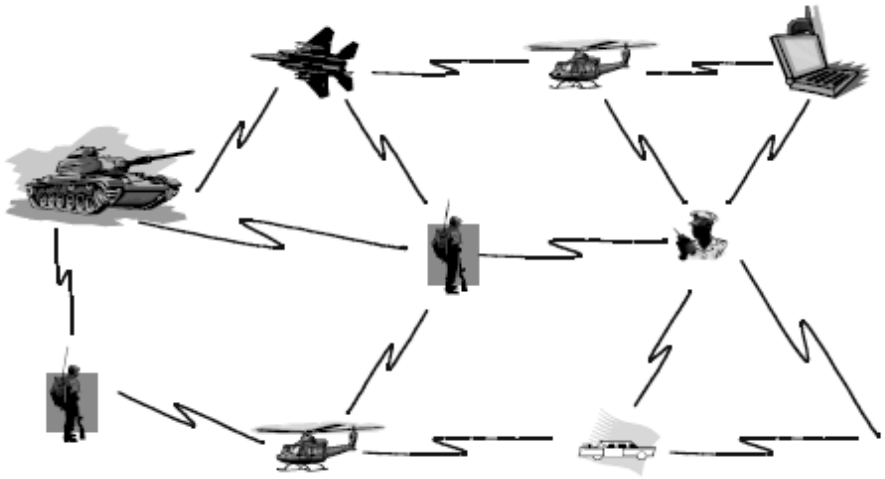
July 2014, 118 Pages**.**

Keywords: AODV , FAODV , AD HOC

1. GİRİŞ

1.1. TASARSIZ AĞ NEDİR?

Tasarsız ağ, bilgiye erişimi dinamik şekilde sağlamak amacıyla plansız olarak birbirlerine bağlanan ve bilgi gönderen kablosuz devrelerin oluşturmuş olduğu kablosuz sistemlerde . Kablosuz devreler, kablosuz LAN kart, Kişisel Dijital Yardımcı (PDA) ya da diğer kablolu veya hareketli iletişim aygıtları tipindeki kişisel bilgisayarlar olabilir (masaüstü/dizüstü). Şekil 1.1 Tasarsız ağın nasıl olduğunu gösterir. Genel olarak, kablosuz devre, havayı iletim aracı olarak gören tüm programlama ekipmanları olabilir. Gösterildiği şekilde, kablosuz devre, aralarında kablolu iletişim sağlamak için herhangi bir kişi, araç ya da uçağa fiziksel olarak bağlantılı olabilir.



Şekil 1.1: Tasarsız Ağ Genel Bakışı.

Tasarsızlarda, kablosuz devre kaynak, varış noktası ya da veri iletim devresi araçları olabilir. Kablosuz bir devre aracı devre görevi gördüğünde, veri paketlerini alabilen ve varış noktasına yakın komşusuna gönderebilen yönlendirici olarak çalışır. Tasarsızın yapısından dolayı, kablosuz ağların oldukları yerde durmaktansa hareket etme eğilimleri vardır. Dolayısıyla, ağ topolojisi zaman zaman değişir.

Kablosuz Tasarsızların bir çok avantajı bulunmaktadır:

Düşük maliyetli yerleşme: Tasarsızlar planlamadan yerleştirilebilirler çünkü bakır kablolar ya da veri kabloları gibi pahalı bir altyapıya gerek yoktur.

Hızlı yerleşme: Tasarsızların yerleşmesi çok pratik ve basittir çünkü yerleşim işleminde hiç kablo yoktur. Yerleşim süresi kısaltılabilir.

Dinamik Yapılandırma: Tasarsız yapılandırması zaman içinde dinamik olarak değişebilir. LAN ayarları ile karşılaştırıldığında, kablosuz bir ağın ağ topolojisini değiştirmek çok basittir.

Tasarsızların çeşitli potansiyel uygulamaları bulunmaktadır. Bazı genel örnekler, acil durum araması-kurtarma operasyonu, toplantı aktiviteleri, konferanslar ve hareket halindeki araç ve/veya askerler arasında savaş alanı iletişimlerini içerir. Hareketli iletişimin yeni taleplerini karşılayabilme yetenekleri ile birlikte, Tasarsızların çok parlak bir gelecekleri bulunmaktadır.

Tasarsız ağlar, sabit bir altyapının ve merkezi sunucuların olmadığı kendiliğinden yapılanan ağlardır. Tasarsız ağlarda düğümler dinamik olarak, süratle ve rastgele yerleşirler. Genelde anlık bir ihtiyacı ya da belirli bir amacı yerine getirmek üzere geçici olarak oluşturulurlar. Her düğüm gerektiğinde yönlendirici olarak çalışır[1][2].

NIST (National Instute of Technology) tasarsız ağları 2 kategoride sınıflandırmaktadır[3]:

1. Gezgin tasarsız ağlar (Mobile ad hoc Networks, MANETs)
2. Duyarga ağları (Wireless Ad Hoc Sensor Networks)

Gezgin tasarsız ağlar sivil uygulamalar için daha çok tercih edilir. Daha ucuz, hafif ve kolay kullanılır olması beklenir. Diz üstü bilgisayarlar, PDA'lar gezgin tasarsız ağ uygulamalarında yaygın olarak kullanılmaktadır. Duyarga ağları ise daha çok askeri, güvenlik ve ölçüm amaçlıdır. MANET'e oranla çok daha fazla düğüm içerir ve aygıtları daha kısıtlıdır. Duyarga ağları başarısız olmaya daha eğilimlidir[4].

1.2. *GÜNCEL* ZORLUKLAR

Tasarsız ağlarda, tüm ağlar paketleri ağa göndermek için birbirleri ile işbirliği içindedir ve bu nedenle devre, etkin bir biçimde yönlendirici görevindedir. Dolayısıyla, en önemli sorunlardan biri yönlendirmedir. Bu tez genellikle Tasarsızlardaki yönlendirme

sorunlarına odaklanmıştır. Bu bölümde, Tasarsızlardaki bazı diğer sorunlar tanımlanacaktır:

Dağıtılmış ağ: Tasarsız ağda sabit ve planlanmış bir altyapının ve kullanıcıların durumlarını takip eden bir olmadan dağıtılmış kablosuz bir ağdır. Bunun anlamı, müşterilerin durumunu korumak için merkezi bir sunucunun gerekiyor olmayışıdır.

Dinamik Topoloji: Devreler hareketlidir, bu sebeple ağ kendini organize eder. Bu yüzden, ağın topolojisi zaman içinde değişir. Bundan dolayı, bu gibi ağlar için tasarlanmış yönlendirme protokolleri de aynı zamanda topoloji değişikliklerine uyum sağlamalıdır.

Güç Bilinci: Geçici bir ağ içindeki devreler genellikle pille çalıştıkları ve düşman arazilerinde yerleştikleri için, katı güç gereklilikleri vardır. Bunun anlamı uygularan protokollerde pil ömrünü korumak önemlidir.

Adres Belirleme Planı: Ağ topolojisi dinamik olarak değişmektedir ve bundan dolayı adres belirleme planı belirgin şekilde kullanılır. Dinamik bir ağ topolojisi, adres kopyalamayı engelleyen yaygın bir adres belirleme planı gerektirir. Kablosuz WAN'lı çevrelerde, Hareketli IP [16] kullanılır. Sabit ev araçları ve yabancı araçlara ihtiyaç olduğu için, bu çözüm Tasarsızlar için uygun değildir.

Ağ Boyutu: Konferans salonları, toplantı vs. yerlerde ses iletimi gibi ticari uygulamaları etkin hale getirme özelliği, Tasarsızların ilgi çekici bir özelliğidir. Ancak, Vurgulanan protokollerdeki gecikme ağ boyutu üzerinde katı bir üst sınır oluşturur.

Güvenlik: Tasarsızlarda güvenlik savaş alanı gibi senaryolarda son derece önemlidir. Güvenliğe ait beş hedefin – hazır bulunurluk, gizlilik, bütünlük, orijinallik ve ret edilmeme- Tasarsızlarda elde edilmesi zordur, bunun nedeni ağ içindeki her devrenin yönlendirme paketlerine eşit şekilde katılmasıdır.

Kablosuz bağlantı kullanımı, Geçici devrelerin pasif gizli dinlemeden aktif canlandırma, mesaj taklidi ve mesaj bozulmaya kadar değişen bağlantı saldırılarına karşı duyarlı olmasını gerektirir. Taklit saldırganın gizli bilgiye erişimini sağlayabilir ve böylece gizliliği ihlal eder. Aktif saldırılar mesajların silinmesi, hatalı mesajların gönderilmesi, bir devrenin bozulmasını vs. dolayısıyla hazır bulunurluk, bütünlük, orijinallik ve ret edilmenin ihlalini içerebilir. Nispeten zayıf fiziksel korumalı düşman bir çevrede özgürce hareket eden devrelerin tehlikeli olma olasılıkları göz ardı edilemez. Bu sebeple, sadece dışarıdan değil aynı zamanda tehlikeli devrelerin içinden gelen kötü niyetli saldırıları da dikkate almalıyız. Yüksek oranda sürdürülebilirlik için,

Tasarsızların savunmasız alanları gösteren merkezi teşekkülleri olmayan dağıtılmış mimarileri olmalıdır. Tasarsızlar topolojideki sık değişikliklerden dolayı dinamiktir. Özellikle bazı devrelerin bozuk olduğu görüldüğünde, bireyler arasındaki güven ilişkisi dahi değişir. Güvenlik mekanizması plansız (dinamik) olmalı ve sabit olmamalıdır ve yüz binlerce devreyi ölçekleyebilmelidir.

1.3. GÜVENLİK HEDEFLERİ

1.3.1. Uygunluk

Ağ hizmetlerinin hizmet aksatma (DoS) saldırılarına rağmen varlığını koruması hedeflenir. DoS saldırıları tasarsız ağların her katmanında gerçekleştirilebilir. Kötü niyetli kullanıcı fiziksel ve MAC katmanlarında ağa mesaj yığarak kanallarda çatışma yaratabilir, ağ katmanında yol atama protokolünün çalışmasını aksatarak ağ bağlantısını koparabilir ve daha üst katmanlarda üst düzey hizmetleri çökertebilir.

1.3.2. Asıllama

Haberleşilen düğümün iddia ettiği düğüm olup olmadığı sınanmalıdır. Bir düğüm başka bir düğümün yerine geçerek yetkisi olmadığı halde kaynağa erişmemeli ve diğer düğümlerin işleyişine müdahale etmemelidir.

1.3.3. Bütünlük

Bu nitelik iletilen mesajın bütünlüğünün bozulmayacağının garanti edilmesini ifade eder. Mesaj kötü niyetli olmayan bozucu bazı etkenlere maruz kalabilir ya da ağda kötü niyetli saldırılara uğrayabilir. Her şartta Verinin bütünlüğü korunmalıdır.

1.3.4. Güvenirlik

Güvenirlik niteliğinde bilginin yetkisi olmayan kullanıcılara açılmaması hedeflenir. Askeri ya da stratejik bilgiler gibi hassas bilgilerin ağda iletimi güvenirliği gerektirir. Bu gibi bilgilerin rakibe sızması hüsranla sonuçlanabilir. Yol atama bilgisinin güvenirliği de bazı durumlarda daha değerli olabilir. Savaş alanında düşman bu bilgiye göre nişan alabilir.

1.3.5. İnkar edememe

Mesajı gönderen düğüm daha sonra göndermediğini iddia edememelidir. A düğümü B düğümünden hatalı bir mesaj aldığında bu mesajı kullanarak B'yi suçlayabilmeli ve diğer düğümleri bu konuda ikna edebilmelidir. Güvenlik konuları [11]'de ayrıntılı olarak anlatılmıştır.

1.4. TEZİN AMACI

Güvenlik politikaları tasarsız ağları için çok önemli bir yer tutmaktadırlar. ilk olarak tezin amacı sağlam bir çevre üzerinde tasarsız oluşturacağız. Bu nedenle güvenlik bir protokol tasarım ulaşacağız, bu yüzden mevcut güvenlik protokol analiz ve avantaj ve dezavantaj araştırdık, son olarak yeni bir mekanizma Karadelik ile Kimlik Doğrulama prosedürü AODV protokolü eklendik.

2. GENEL KISIMLAR

2.1. GİRİŞ

Tasarsız ağlar, sabit bir altyapının ve merkezi sunucuların olmadığı kendiliğinden yapılanan ağlardır. Tasarsız ağlarda düğümler dinamik olarak, süratle ve rastgele yerleşirler. Genelde anlık bir ihtiyacı ya da belirli bir amacı yerine getirmek üzere geçici olarak oluşturulurlar. Her düğüm gerektiğinde yönlendirici olarak çalışır[1][2].

Kablosuz ağlar, bilgiyi hızlı sağlamak, uzaklık ve zaman engellerinin ortadan kaldırılması gibi çok farklı alanlardaki uygulamalarda (örneğin felaket kurtarma, yangın, sel, deprem v.b.) askeri haberleşmede (komut, kontrol gözetleme ve keşif v.b.) kullanılırlar. [3][9].

Tasarsız özel bir ağ, kablosuz, mobil, kurulmuş herhangi bir altyapı veya merkezi yönetimin yardımı olmadan geçici bir ağ oluşturarak oluşur.

Tasarsız ağların seyyar tasarsız ağlarda olduğu gibi askeri ve sivil uygulamalarda potansiyelini çalışabılmesi için binlerce emniyet engelini yenmeleri gerekmektedir. Genelde tasarsız ağlar güvensiz çevreler içinde yayın yapar. Bu nedenle güvenirlik bu ağlardaki emniyetli ilişkilerin habercisidir. Gönümüzde tasarsız ağlar için bir çok güvenirlik protokolleri önerilmiştir. Şu an, bu protokolleri değerlendirmek için bir genel çerçeve yoktur. Böyle bir yapı geliştirmek için, bu araştırma, genel güvenilirlik metodu ve bu literatürde bildirilmiş protokoller arasındaki farklılıkları ve benzerlikleri açıklayan yeni bir taksonomi önerir. Bu taksonomi güvenilrlik fonksiyonundaki düğümlerin rolüne, protokol kurulmasına ve protokolün tipine dayanır.[9][4]

Tasarsızların kendilerini bazı saldırı türlerine karşı savunmasız bırakan belirli özgün özellikleri bulunmaktadır. Tüm devreler ağ içine paket gönderiminde yardımlaştıkları açık bir çevre içine yayılmış olduklarından, kötü niyetli devrelerin tespit edilmesi zordur. Dolayısı ile, kablolu ya da altyapıya dayalı kablosuz ağlarla karşılaştırıldığında bir güvenlik protokolü tasarlamak çok zordur. Bu bölüm, bir güvenlik protokolleri tasarımcısının yüz yüze geldiği bazı sorun ve zorlukları tartışır. Sorunlar bir güvenlik protokolünün öncelikli hedefleri ile ilgili olarak analiz edilmiştir – gizlilik, bütünlük

hazır bulunurluk, orijinallik ve ret edilmeme durumu. Var olan Tasarsız yönlendirme protokolleri tarafından izin verilen saldırı ve tehditler daha sonra tartışılmıştır. Akabinde bir kaç güvenli yönlendirme protokolünün çalışması tanımlanmıştır. Bir sonraki bölüm Tasarsızlardaki bir diğer önemli sorun olan - sertifikaya dayalı orijinalliği - tartışır. Bu bölümace tasarsızlarda etkili sertifikaya dayalı orijinallik gerekliklerini sunan ve analiz eden bazı mekanizmaları irdelenmiştir.

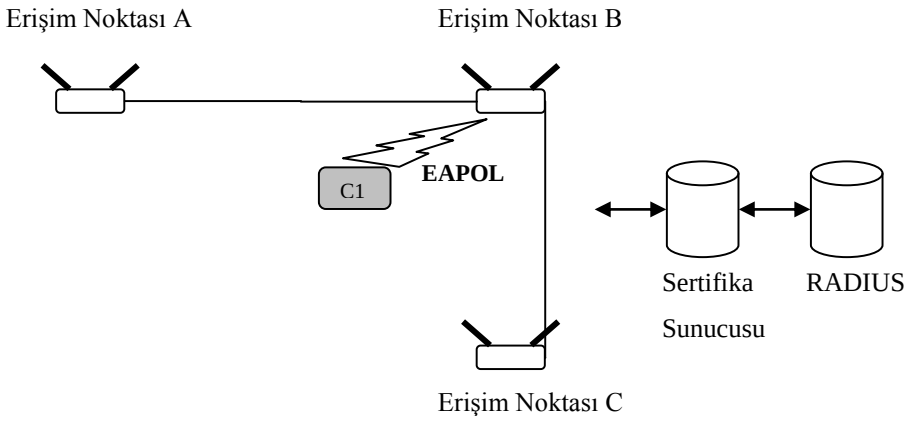
2.2. GÜVENLİ PROTOKOL TASARLAMADAKİ ZORLUKLAR

Bu bölüm Tasarsızlar için güvenli protokolr tasarlamadaki sorun ve zorlukları sıralar [1].

Ortak yayın radyo kanalı: Tasarsızlarda, radyo kanalı tüm devreler ile veriyi yayınlayarak paylaşılır. Bu sebeple, herhangi bir kötü niyetli devre veri üzerine kolayca "girebilir", dolayısı ile ağ gizliliğini ihlal eder. Bu durum yönlü anten kullanılarak en aza indirgenebilir.

Düşman Ortam: Tasarsızlar genellikle savaş alanı gibi düşman ortamlarda yayılmıştır. Böyle durumlarda, devrelerin kendileri saldırılara eğilimlidir. Dolayısıyla, sadece protokol adresi ağ dışından saldırıda bulunmaz, ayrıca saldırılar ağın içinden de başlatılır. Bunun gibi iç saldırılar şiddetlidir ve bütün güvenlik amaçlarını ihlal edebilir.

Dağıtılmış Mimari: Kablolu ve kablosuz altyapılı ağlarda (Kablosuz LAN gibi), kullanıcıların kimliklerini doğrulayan ve anahtar yönetimiyle ilgilenen sabit bir hat vardır. Örneğin, kimlik doğrulama için bir RADIUS **(**Remote Authentication Dial In User Service) sunucusu ve anahtar yönetim için bir sertifika sunucusu şekil 3.1'de tarif edildiği şekilde kurulmuştur. Burada, kablosuz istasyon C1, bilgiyi destek kimlik doğrulama sunucusuna gönderen EAPOL (The Extensible Authentication Protocol) gibi bir protokol kullanarak kimlik doğrulama yeterlilik belgesini kimlik denetleyicisine gönderir (Erişim Noktası B).



Şekil 2.1: Kablosuz LAN'da Merkezi kimlik doğrulama şeması.

Adı geçen merkezi şema dağılmış yapısına göre uygun olmayan geçici bir ağdır.

Dinamik Topoloji: Tasarsız içindeki devreler yapısı gereği yüksek oranda dinamiktir ve bu sebeple ağın topolojisi sürekli değişmektedir. Bu nedenle, devreler arası güven ilişkisi de değişmektedir. Örneğin, bir ağ içinde herhangi bir kötü niyetli devre saptanırsa, çevresindeki devrelerle olan ilişkisi değişir. Bu sebeple, güvenlik protokolü de bu değişikliklere uyum sağlamalıdır.

Güç sınırlaması: Piller içinde çalışan PDA, laptop vs. gibi Tasarsız içindeki devreler Güvenliğe ek olarak "üst katmanlar" da çalışan protokoller genel olarak daha fazla performans ister ve aynı zamanda ağ bant genişliğini tüketir. Bu durumun çıkarımlarından biri, devrelerin Hizmet Engelleme (DoS) gibi saldırılara karşı kolayca savunmasız kalmasıdır. Dolayısıyla, güvenlik çözümü de aynı zamanda güç bilinçli olmalıdır.

2.3. TASARSIZLARDAKİ DAĞITIMIN GÜVENCEYE ALINMASI

Bu tez öncelikle bir ağ katman perspektifindeki güvenlik sorunlarına odaklanır. Tasarsızlar için çeşitli dağıtım protokolleri bulunmaktadır, bunlardan hiçbiri güvenlik adı altındaki en önemli soruna değinmez. Saldırı ve tehditleri incelemek ve bu konulara değinen protokolü düzenlemek için, işletim sistemi ile ilgili bir anlayışa ihtiyaç vardır.

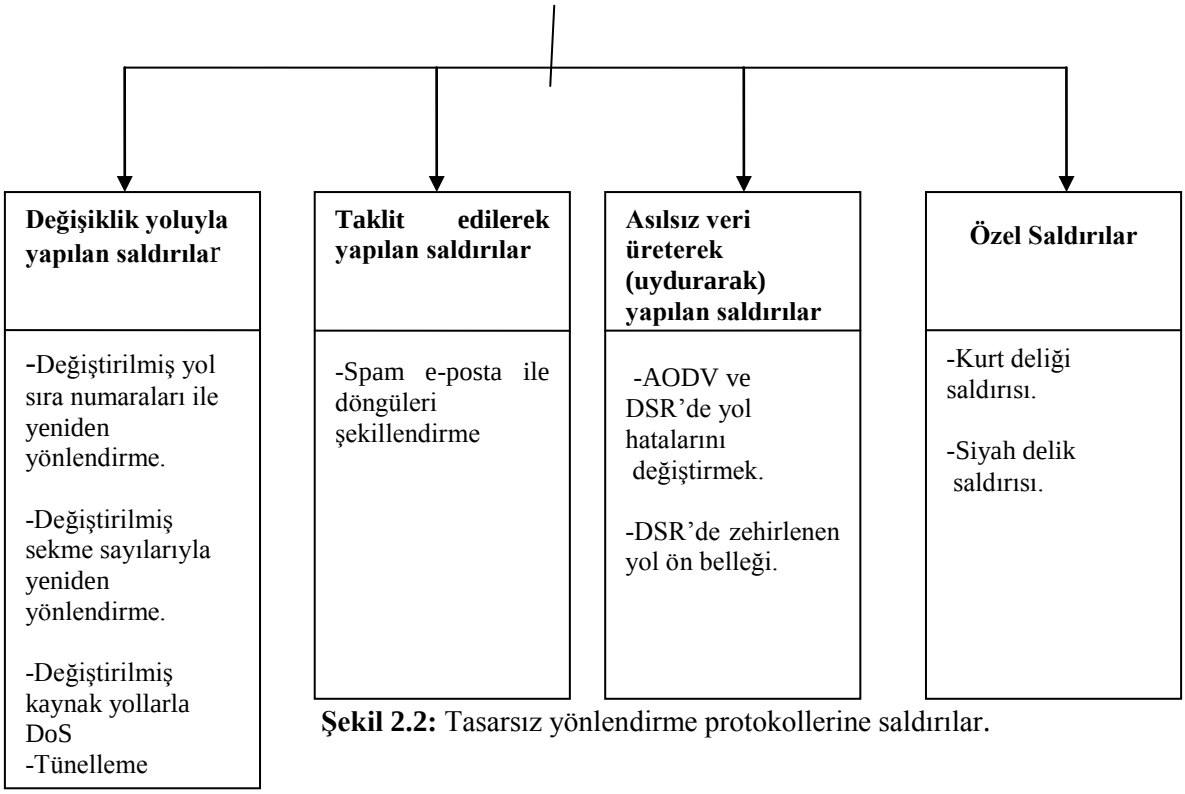
İşletın sistemi, RADIUS sunucusu gibi güvenilir genel bir makamın var olduğu yönetilen bir sistem olabilir veya devreler arasında muhtemel bir güven ilişkisinin olmadığı açık bir sistem olabilir. Örneğin, bir savaş alanı içinde devrelerin anahtar yönetim işlevlerini yerine getira genel bir güvenlir makamı bulunmalıdır.

Tasarsızlar genel olarak açık çevre tipine girerler, çünkü devreler düşünülür ve dinamik olarak bir bağlantı kurarlar. Çevrenin bir diğer olası tipi de, devrelerin hali hazırda bazı güvenlik altyapıları kurdukları *yönetilen-açık* çevrelerdir. Bu durum, devreler arasında bir güven ilişkisi kurmak için başlangıç noktası olarak hareket eder. Çeşitli sertifikaya dayalı mekanizmalar, adı geçen çevreyi olumlar şekilde bölüm 2.4'te tartışılacaktır. Buna ek olarak, çevre, güvenliğin birinci derecede önemli olduğu askeri ağlar gibi senaryoların tanımladığı *yönetilmiş-düşman* olabilir.

Bu çevrelere bağlı olarak, mevcut yönlendirme protokolleri tarafından izin verilen çeşitli saldırı ve tehditler bulunmuştur. Var olan yönlendirme protokolleri bu saldırılara değinmek için geliştirilmiştir. Aşağıdaki bölümler adı geçenleri daha detaylı bir şekilde tartışır:

2.3.1. Mevcut Yönlendirme Protokollerine Saldırı ve Suiistimaller

Yönlendirme protokollerine yapılan saldırılar genellikle *yönlendirme bozulma* saldırıları (saldırgan yönlendirme paketleri ile yanlış yönlerde yönlendirme mekanizmalarını bozmaya çalışır) ve *kaynak tüketimi* saldırıları (bazı işbirliksiz ya bencil devreler ağ bant genişliğini yok etmek için yanlış paketler eklemeyi deneyebilir) olarak sınıflandırılabilirler. Her iki adı geçen saldırı da Hizmet Engelleme (DoS) örnekleridir. Şekil 2.2 Tasarsızlardakı olası saldırıların daha geniş bir sınıflandırmasını tanımlar.



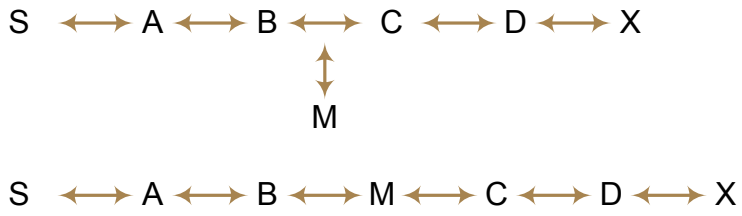
Şekil 2.2: Tasarsız yönlendirme protokollerine saldırılar.

Şekil 2.2: Tasarsız yönlendirme protokollerine yapılan saldırıların sınıflandırılması. Tanımlanan saldırıların çoğu AODV, DSR, vs. gibi talepli ve reaktif protokollere odaklanmıştır. Aşağıdaki bölümler bu saldırılara daha detaylı bakmaktadır.

2.3.1.1. Değişiklik Kullanan Saldırılar

Bu saldırı tipinde, devreler arasında geçen mesajların protokol alanları değiştirilmiştir, dolayısı ile trafik bozulması ya da Hizmet Engelleme (DoS) ile sonuçlanır. Aşağıdaki bölümler bu saldırıların bazılarını tartışır-

(a) *Değiştirilmiş yol sıra numaraları ile yeniden yönlendirme*: Bu saldırı AODV protokolüne karşı olabilir. Şekil 2.3'de gösterilen ağı dikkate alalım [13]. M, S kaynaklı varış yeri X olan bir yol üzerinde RREQ paketini gizlice dinleyen kötü niyetli bir devre olsun. M X'e daha uzun bir yolla yanlış RREP paketi (kendisini de listeye ekleyerek) ve X ile son olarak tanıtılan adı geçenden daha büyük bir varış sıra numarası gönderir. Bu da, S'nin tüm paketlerini M aracılığıyla yönlendirmesini sağlar çünkü M, X'e daha yeni bir yön tanıtmış olur.



Şekil 2.3: Kötü niyetli devreli bir Tasarsız Ağ örneği [13].

(b) *Değiştirilmiş sekme sayılarıyla yeniden yönlendirme:* Bu saldırı türü, kötü niyetli bir devreye ait RREQ paketinin sekme sayısını sıfırlayarak tekrar başlatan yeni oluşturulmuş bir yolun içerilme şansını çoğaltabilen bir AODV protokolüne karşı hedeflenmiştir.

(c) *Değiştirilmiş kaynak yollarla Hizmet Engellemesi*: Bu saldırı, kaynak yol kullanan ve yukarıdaki şekilde çalışan DSR'ye karşı olasıdır - şekil 2.3'de en kısa yolun S'den X'e kadar mevcut olduğunu varsayın. Aynı zamanda C ve X'in birbirlerini duymadığını ve B ve C devrelerinin birbirlerini duymadığını ve M'nin hizmet engelleme saldırısını deneyen kötü niyetli bir devre olduğunu varsayın. S'nin S-A-B-C-D-X devre yolu ile birlikte S'ye bir veri paketi yolladığını düşünün. M, bu paketi engeller ve D'yi listeden çıkarıp Cye gönderirse, C bu paketi X'e göndermeye çalışacaktır. Fakat C, X'i

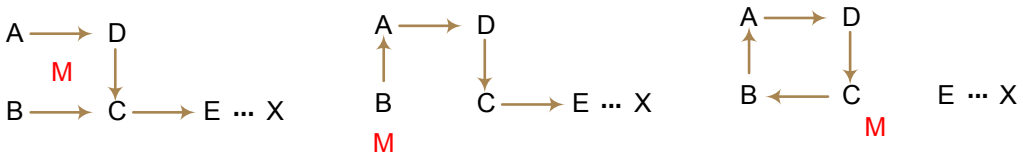
duyamadığı için bu durum imkansızdır. Dolayısıyla, M, X üzerinde başarılı bir DoS saldırısı başlatmıştır.

2.3.1.2. Canlandırma Kullanan Saldırılar

Bu saldırı çeşitleri ağ içindeki orijinallik ve gizliliği ihlal eder. Kötü niyetli bir devre, başka bir devre tarafından algılandığı şekilde ağ topolojisi görüntüsünü değiştirmek için bir diğer devrenin adresini taklit edebilir ya da değiştirebilir. Bu şekilde saldırılar aşağıda tanımlandığı gibi döngülerin oluşmasına sebep olabilirler -

(a) Spam e-posta ile döngüleri şekillendirme: Şekil 2.4 [13]'te gösterildiği şekilde, yolun A'dan X'e gittiğini farz edelim. Ayrıca, A'nın B ve D'yi duyabildiğini (bunun anlamı A'nın B ve D radyo frekansında olmasıdır), B'nin A ve C'yi duyabildiğini, D'nin A ve C'yi duyabildiğini ve C'nin B,D veE'yi duyabildiğini farz edelim. Aynı zamanda kötü huylu bir devre olan M'nin A, B, C ve D'yi duyabildiğini düşünelim. Bu saldırı aşağıdaki şekilde tanımlanabilir:

"Saldırıya başlamak için, M, A ile eşleşmek amacıyla MAC adresini değiştirir, daha yakın ve düğümünü kapsama aralığı dışında hareket eder. Daha sonra, C ile gönderilenden daha az şekilde X'te bir sekme sayısı içeren B'ye bir RREP gönderir. Dolayısıyla, şekil 2.4. de gösterildiği gibi B, ağda ilerlemek için varış yeri X olan yolu değiştirir. Daha sonra B'ninki ile eşleşmek için MAC adresini değiştirir akabinde B düğümünü kapsama aralığı dışında kalacak şekilde daha yakın bir böyle hareket eder E ile tanıtılandan daha düşük şekilde sekme sayısına sahip ile bir RREP mesajını C'ye gönderir. C de, B aracılığı ile X'e yönelir. Bu noktada, bir döngü oluşmuştur ve X dört devreden de ulaşılamayacak bir pozisyonda düşmüştür. Saldırı tek bir kötü huylu saldırgan ile yapılabilir; ancak çoklu saldırganlar aynı sonuç için birbirlerine yardım edebilirler."



Şekil 2.4: Spam e-posta ile döngüleri şekillendirme [13].

2.3.1.3. Üretim Kullanan Saldırılar

Bu saldırı şeklinde, kötü niyetli bir devre yönlendirme mekanizmasını engellemek için sahte mesajlar ve yönlendirme paketleri gönderilmeye çalışılır. Bu saldırıları bir Tasarsız içinde fark etmek zordur. Yönlendirme paketleri, kendilerini işleme tabi tutan devrelerde geçerli paketler olarak görülür. Aşağıdaki saldırılar üretim ile yapılan saldırılara örnektir [13].

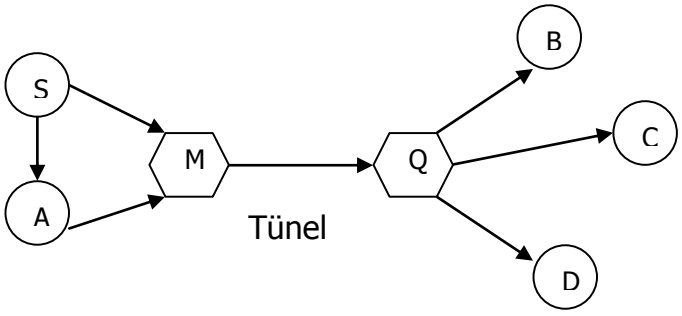
(a) *Yol hatalarını değiştirmek*: Bu türdeki saldırılar, AODV ve DSR yönlendirme protokolleri içine (Bölüm 2'de tanımlanmıştır) yol hatası paketleri (RERR) göndermek için olan yayım mekanizmasını kullanır. Aynı şekilde Şekil 2.3.'te gösterilen ağları göz önünde alamuzde. M'nin B ve C'den gelen yayım paketlerini gizlice dinleyen kötü niyetli bir devre olduğunu farz edilim. M, C ve X devreleri arasında hatalı bir bağlantı gösterir şekilde devre B'de yanlış RERR paketi yayınlayarak X'e karşı bir üretim saldırısı başlatabilir. B, C'den geldiğini düşünerek dolandırıcı bir yol hata mesajı alır. Dolayısıyla, devre X'teki yönlendirme girişini siler ve RERR paketini aynı şekilde uygun durumu takip eden devre A'ya gönderir. Devre S, devre X'e doğru yol kurduğu zamanlarda, M bu prosedürü tekrar eder. Bu da devre S'nin devre X ile iletişime geçmesini engeller.

(b) *Zehirlenen yol ön belleği*: Hatalı yönlendirme paketleri ekleyerek yönlendirme tablolarını değiştiren ya da bozan saldırılarla ilgilidir. Adı geçen saldırı, devrelerin komşu devrelerden gelen yolları yayın kanalından fark gözetmeksizin dinleyerek fark ettiği DSR protokolünün optimize edilmiş bir sürümüne karşı olasıdır. Örneğin, şekil 2.3'te, herhangi bir düğüm S'den X'e giden yolu gizlice dinlerse, kendi yönlendirme tablosuna bir giriş ekleyecektir. Kötü niyetli bir M devresinin, kendisi aracılığıyla X'e yol tanıttığını düşünürsek, (örn. S-A-B-C-M-C-D-X) , devrenin yol tablosunu dinlerken hatalı yol girişleri oluşu.

2.3.1.4. Özel Saldırılar

Yukarıda tanımlanan saldırılar dışında, AODV, DSR vs. gibi yönlendirme protokollerine karşı mümkün olabilen şiddetli iki farklı saldırı bulunmaktadır. Aşağıda tanımlanmışlardır -

(a) Kurt deliği Saldırısı: Kurt deliği saldırısı, iki adet gizlice sızan kötü niyetli devrenin bir "tünel" aracılığıyla paketleri tünelle gönderebildiği ya da şekil 2.5'te gösterildiği şekilde ağlar içinde tepe noktasıyla kesiştiği şiddetli bir saldırı türüdür.



Şekil 2.5: Kurt deliği saldırısı.

Burada, M ve Q, paketleri bir alt ağdan diğerine tünelle gönderen iki kötü niyetli devredir. Bu gibi bir saldırı türünü bir ağ içinde tespit etmek zordur ve bu saldırı tipi devreler arasındaki iletişime ciddi şekilde zarar verir. Bu saldırılar, ağdaki hatalı paketleri belirlemek için paket içindeki zamanlama bilgisini doğrulayan bir *paket bağı* kullanarak önlenebilir. [14].

(b) Siyah delik saldırısı: Bu saldırı tipinde,çevresindeki tüm devrelerin kendisine doğru paketle yönlenmesine sebep olur şekilde, bir devre tüm varış noktaları için sıfır metre ölçümü tanıtır. AODV protokolü bu şekilde bir saldırıya karşı savunmasızdır. Bu saldırı ile ilgili daha fazla detay [15]'te bulunabilir.

2.3.2. Tasarsız Yönlendirme Protokollerindeki Saldırıların Risk Değerlendirmesi

Tablo 2.1, beş güvenlik hedefinin ihlali ile ilgili saldırıların bir risk değerlendirmesini verir. Saldırılar, 1'in anlamının saldırının önemsiz bir etkisi olduğu ve 5'in anlamının saldırının şiddetli bir etkisi olduğu, 1-5 ölçeği üzerindeki önem derecesine dayanır şekilde oranlanmıştır.

Tablo 2.1: Tasarsız yönlendirme protokolleri üzerindeki saldırıların risk değerlendirmesi.

Hedefler Saldırı	**Gizlilik**	**Veri Bütünlüğü**	**Kullanabilirlik**	**Orijinallik**	**Ret edilmeme**
Değişiklik kullanan saldırılar	1	5	5	1	2
Canlandırma kullanan saldırılar	3	4	5	5	5
Üretim kullanan saldırılar	1	5	5	3	5
Kurt deliği Saldırısı	4	4	5	4	4
Siyah delik Saldırısı	2	3	5	1	1

Tablodan çoğu saldırının kullanılabilirliğin güvenlik hedefini olumsuz olarak ihlal ettiği sonucu çıkarılmıştır. Tasarsız içindeki devreler kaynak sınırlı olduğu için bu durum özellikle önemli olabilir ve değişimle yapılan bu şekildeki saldırılar, kaynakları kolayca tüketebilir. Tuzak devre çiftleri tüm ağ işletimini etkileyebildiği için kurt deliği saldırısı en şiddetli saldırıdır. Değişim ve üretimden kaynaklanan saldırılar devrelerin yönlendirme tabloları içindeki bilgileri değiştirdikleri için genellikle bütünlüğü ihlal ederler. Dolandırıcı adresli herhangi bir devre orijinal devrenin eşlerinin kimliğini doğrulamayı engellediği için orijinallik canlandırmasından kaynaklanan saldırıları olumsuz yönde etkiler.

Güvenli bir protokol ve saldırıları tasarlama zorlukları ile ilgili bir tartışmadan sonra, bir sonraki bölüm geçici ağlar için bir kaç güvenli yönlendirme protokolünü tartışır.

2.4. TASARIZLARDA SERTİFİKAYA DAYALI KİMLİK DOĞRULAMA

Sertifikaya dayalı kimlik doğrulama kablolu ağlarda oldukça çalışılmıştır. Ancak, basit olmayan bir görevde, sertifikaya dayalı kimlik doğrulama protokollerini hareketli geçici ağlara (Tasarsızlar) uyarlamanın asıl nedeni, geleneksel kablolu ağların aksine bir Tasarsızda sabit bir altyapı ve merkezi yönetiminin bulunmamasıdır. Örneğin, geleneksel bir sertifikaya dayalı kimlik doğrulama sistemi, sertifikaların oluşturulması, dağıtımı, yenilenmesi ve iptalinden sorumlu sabit bir güvenilir Tasdik Yetkisine (CA) dayanır. Bir Tasarsızda, hareketlilik, sınırlı kablosuz kanal, sıkça görülen bağlantı hataları gibi sorunlardan dolayı, ağ içine adı geçen sabit merkezli bir CA dahil etmek genellikle elverişli değildir. Çeşitli yaklaşımlar, hareketli ve geçici ağlarda dağıtılmış doğrulama için sertifikaya dayalı yöntemleri uyarlamanın benlirtilen zorluğunu azaltmayı hedeflerimizledir. Bu noktada bu tezin iki hususta katkı sağlayacağı gösterebilir ilk olarak Tasarsızlar için güvenli dağıtılmış doğrulama sistemlerinin gereklilik analizi yapılmıştır. İkinci olarak dağıtılmış doğrulama içeriğinde halihazırda var olan sertifikaya dayalı doğrulama mekanizmalarının, avantaj ve dezavantajlarını içeren karşılaştırmak bir araştırmanın yapılmış olmasıdır.

2.4.1. Sorunlar

Güvenlik mekanizmalarının beş öğesi, gizlilik, bütünlük, orijinallik, kullanabilirlik ve ret edilmeme durumudur. Bunların dışında, doğrulama ihlali sistem çapında riskli anlaşmaya yol açtığı için, doğrulama en önemli temel sorundur. Geleneksel kablolu ağlarda geniş çapta kullanılan doğrulama mekanizması sertifika kullanan genel anahtar yönetim sistemidir. Sertifikaya dayalı yöntemde dikkate alınan temel sorunlardan biri, genel anahtarın ağ içindeki tüm devrelere güvenli dağıtımıdır. *Genel Anahtar Altyapısı (PKI)* [20], X.509 sertifikaları kullanarak genel anahtar yönetimini idare etmek için yöntemler tanımlar. Kablolu ağlarda, sertifikaların oluşumu, yenilenmesi ve iptalini yöneten merkezi bir sertifika sunucusu bulunur. Sabit altyapı ve merkezi yönetimin olmayasından dolayı bu durum geçici ağlarda uygulanamaz. Ayrıca, ağın dinamik topolojisinden dolayı, tekrar doğrulama ve sertifika sunucusu ile periyodık iletişim gibi sorunlara sebep olan sık bağlantı hataları görülebilir.

Bu sınırların üstesinden gelmek ve sertifikaya dayalı doğrulama mekanizmasının tüm avantajlarından yararlanmak için, bir kaç genel anahtar yönetim mekanizması önerilmiştir [21] [22] [23] [24] [25] [26]. Aşağıdaki bölümler bu yöntemlerden bazılarını analiz eder ve yöntemlerin avantaj ve dezavantajlarını tartışır. Bölümün geri kalanı aşağıdaki şekilde organize edilmiştir. Bölüm 2.4.2, hareketli ve geçici ağlarda sertifikaya dayalı doğrulama şeması gerekliklerini tartışır. Bölüm 2.4.3, kullanılmış mekanizmaların genel tanımını ve konu ile ilgili bir araştırmaya kapsar. Bölüm 2.4.4'te gerekliliklerle ilgili şema karşılaştırılması sağlanır.

2.4.2. Gereklilikler

Hareketli geçici ağda onaylama ile ilgili olarak *güvenli ve etkili* görülen herhangi bir sertifikaya dayalı onaylama şeması için beş gereklilik tanımlanmıştır.

1: *Dağıtılmış onaylama*: Geçici ağlarda sık bağlantı hataları, devre hareketliliği ve kablosuz ortamın erişim sınırlamaları gibi sorunlardan dolayı, genellikle ağ içine sabit merkezi Tasdik Yetkisi (CA) dahil etmek uygun değildir. Ayrıca, yüksek güvenlik gerektiren ağlarda, adı geçen sunucu hatanın tek noktası olabilir. Örneğin, askeri birliklerin geniş bir alana yayıldıkları bir savaş sahası senaryosunu düşünelim. Böyle bir durumda, merkezi bir sunucuya sahip olmak uygun olmayabilir. Adı geçen sunucuda bir düşman saldırısı düşünün - bu saldırı tüm ağı yıkabilir! Sertifikaya dayalı

mekanizmaların temel gerekliliklerinden biri ağ içinde devre dizileri arasında onaylama dağılımı yapmaktır.

2: *Kaynak Bilinçli*: Bir geçici ağ içindeki devreler genellikle yüksek güç tüketimli ve düşük hafıza kapasiteli pillerle çalıştığı için, onaylama protokolleri kaynak bilinçli olmalıdır. Bunun anlamı, temel algoritmalarının zaman ve mekan karmaşıklığı kabul edilir şekilde düşük olmasıdır. Bu bağlamda, genellikle simetrik kriptografi daha az kaynak tüketimine maruz kaldığı için, simetrik anahtara bağlı teknikler genel anahtar yöntemleri ile karşılaştırıldığında daha uygundur. Ancak, simetrik anahtarların dağılım sorunu geçici ağlar içindeki pratik yayılımlarını önler. Bu da, uygulama seviyesi ile işlem görmesi gereken dengedir. Sertifikaya dayalı doğrulama kaynağa yoğunlaşmış genel anahtar mekanizmaları kullandığı için, protokolün kendisi hem bellek ve hem de güç yönünden etkili olmalıdır.

3: *Etkili sertifika yönetim mekanizması:* Genel anahtar ve sertifika yönetimi dağılımı kablolu ağlar için yoğun olarak çalışılmıştır [20]. Ancak, bu yöntemleri Tasarsızlara uygulama, sertifikaların yönetiminin zorlaşması anlamına gelir (oluşturma, iptal ve yenileme). Bu sorun daha sonra Bölüm 3 ve 4'te tartışılmıştır. Halihazırdaki mekanizmaların çoğunda güvenli bir sertifika iptal şeması eksiktir.

4: *Çok türlü sertifikasyon:* Kablolu ağlarda olduğu gibi, Tasdik Yetkileri geçici ağlarda da çok çeşitli şekil ile gerçeklerbilir farklı "etki alanına" ait iki ya da daha fazla devrenin birbirlerini doğrulamaya çalışabilecekleri anlamına gelir. Böyle bir durumda, Tasdik Yetkileri arasında bir çeşit güven ilişkisi ya da hiyerarşi olması gerekmektedir. Kablolu ağlarda, bu durum sertifika zincirleme yoluyla başarılır.

5: *Ön doğrulama mekanizmasının güvenliği:* Ön doğrulama mekanizması, mevcut sertifika oluşumu ve dağılımından önce devreler arasında gerekli güveni sağlama süreci ile ilgilidir. Bu durum, sertifika doğrulama sürecinin kendisine ait bir bölüm olmamasına rağmen, Tasarsızlarda çok önemlidir. Bunun birinci sebebi bağlantıyı

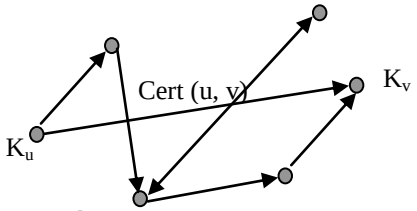
tamamlamak için, devrelerin arasında önceden güven sağlanmış olması zorunluluğudur (örneğin, genel anahtar değişimi ile). Güven sağlanmadan, sonraki karşılıklı sertifika onayı ve yenilenmesi mümkün değildir. Stajano and Anderson [27] tarafından önerilen Yavru Modeli Yenilenme, yer ile sınırlı kanal üzerinde "anne" ve "yavru" devre arasındaki güven sağlamayı içeren bu alanda yapılmış eski çalışmalardan biridir. Balfanz et al [28] daha kullanıcı dostu ve etkili bir yaklaşımı tartışır.

2.4.3. Literature Taraması

Sertifikaya dayalı onaylama genellikle üç aşamadan oluşur. İlk aşama ya da "ön yükleme" aşaması sırasında, devreler tasdik yetkisi ile bir sertifika verirler. Bu sertifika, IP adresi, isim, organizasyon ve genel anahtar gibi devre kimlik bilgileri kullanarak Tasdik Yetkisi tarafından oluşturulur. Sertifika aynı zamanda diğer bilgilerin dışında zaman ve vade süresi vermekten zorundadır. İkinci aşama sırasında, sertifika vade sonu sebebiyle "yenilenmiştir". Üçüncü aşama, muhtemelen sertifika sahibinin genel anahtar uyuşmasından kaynaklanan ya da yayınlayanın kullanıcı anahtarı bağlayıcısının artık geçerli olmayışı sebebiyle sertifikanın Tasdik Yetkisi ile iptalini içerir. Önerilen bazı mekanizmalar aşağıdaki alt bölümlerde anlatılmıştır.

2.4.3.1. Kendini Organize Eden Genel Anahtar Yönetimi

Capkun, Buttyan ve Hubaux tarafından önerilmiş sertifikaya dayalı doğrulama yöntemlerinden biri, sertifika grafiği oluşturulmasıdır [21]. PGP içinde merkezi bir sertifika kullanıldığı gerçeği dışında, önerilen yaklaşım PGP sertifikalarına benzerdir [29]. Sertifika grafiği, V ve E'nin sırasıyla düğüm ve bağ dizisini temsil ettiği yönlendirilmiş bir grafik G (V, E) olarak tanımlanır. Sertifika grafiğinin düğüm noktaları genel anahtarları ve bağler de sertifikaları temsil eder. Şekil 2.6'da gösterildiği gibi, K_u - K_v arası bağden gelen grafik içindeki yönlendirilmiş bağ, kendi özel anahtarı ile birlikte v'nin genel anahtarı K_v ile u tarafından imzalanan Cert(u, v) ile birlikte verilmiş sertifikayı temsil eder. Dolayısıyla bu aslında, , u, v için Tasdik Yetkisidir. G sadece tüm ağın geçerli sertifikalarını içerir.



Şekil 2.6: Ku → Kv, v tarafından u'ya verilmiş sertifika.

Her devre, sırasıyla güncelleştirilmiş ve vade sonu gelmiş sertifikaların alt kümesinden oluşan güncellenmiş ve güncellenmemiş yerel sertifika belleğini muhafaza eder. Capkun et al, iyi bir sertifika grafiği tahmini ve devre doğrulaması sağlamada iki belleğin kullanımını tartışır. Bir u kullanıcına diğeri v kullanıcına ait genel anahtarı doğrulamak istediği zaman, u, u ve v'nin güncellenmiş sertifika bellek grafiğini birleştirerek grafik içinde yönlendirilmiş bir yol bulmaya çalışır. Yol üzerindeki sertifika zincirleri v'yi doğrulamak için kullanılır. Hiçbir yol bulunmuyorsa, o halde, devre yolda vade sonu gelmiş sertifika bulmak için güncellenmemiş ve güncellenmiş sertifika belleğini birleştirir. Bu şekilde bir yol bulurken, vade sonu gelmiş sertifikayı günceller, doğruluğu kontrol eder ve doğrulamayı uygular.

Sertifika oluşturma aşaması, kendine ait-özel anahtar çiftlerini üreten her devre ile başlar. Yeni bir devre komşudan yeni bir sertifika talebinde bulunduğunda, sertifikayı veren genel anahtarın orijinalliğini doğrular. Capkun et al, bu işlemin yan bir kanal üzerinden anahtarlarını önceden değiştirme ile yapıldığı varsayımında bulunur. Güncellenmiş bellek içinde sertifika grafiğini güncellemek için, bir sertifika değişim aşaması periyodik olarak komşu devrelerle sertifika karması değişimi ile yürütülür. Tüm devrelerle sertifika grafiği ile güncellenmeden önce yakınsama zamanı üzerinde bir üst bağ bulunmaktadır. *Güncellenmiş sertifika bellek* oluşumu ve güncellenmesinin etkinliğini arttırmak için, Capkun et al, en fazla sertifika sayısı ile sertifika grafiğinde yol bulmaya dayanan *Maksimum derece algoritması* gibi algoritmalar önerir.

Capkun et al, bir devre kendi güncellenmemiş sertifika belleği içinde süresi dolmuş sertifikalar bulduğunda nasıl bir sertifika yenilenme mekanızması kullanılacağına cekinmez. Sertifikaların belleği için net ve gizli olarak iki yöntem önermişlerdir. *Gizli* mekanizmada, sertifikalar süre dolum tarihine dayanarak iptal edilir. *Net* yöntemde, sertifikayı veren, artık geçerli bir kullanıcı anahtar bağlantısı bulunmadığına inanılan net bellek beyanı gönderir. Bu beyan, sertifika verenin hedef devre için sertifika güncellemesi talep eden devrelere gönderilir.

Bu mekanizmanın avantajı, sertifika kullanan genel anahtarların tam olarak kendi kendine organize olabilme özelliğinde saklıdır. Ancak, bu mekanizmanın zorluğu sertifika bellekleri için tutulması gereken tablolardır. Bir devre, bir yerden diğerine hareket ederse, diğer devrelerle yeniden görüşmeli ve tabloları tekrar güncellemelidir.

2.4.3.2. Tasarsızlar İçin Sağlam ve Yaygın Güvenlik Desteği Sağlamak

Bu yönetimde, Kong et al [23], giriş kriptografisi ve paylaşımlı gizli anahtara sırlara dayanan dağıtılmış bir sertifikasyon kullanır. Giriş gizlianahtar paylaşım yönteminin temel amacı, gizli bir polinom f(x) kullanan, isteğe göre geniş olan bir grup arasında gizli anahtar *k* kullanmaktır. Eğer f(x) lin derecesi (*k*-1) ise, *k*'dan az üyeler gizlianahtar ile ilgili bir bilgi ortaya koymaz iken, grubun herhangi bir *i* üyesi gizlianahtar değerlendirebilir. Burada, *k,* yöntemin etkili olması için dikkatli bir şekilde ayarlanmış olması gereken bir parametredir.

Sertifika oluşum süresi aşağıdaki şekildedir: İlk olarak ağ içindeki tüm devrelerin güvenilir bir merkezi birimden sertifikaları ile ön yükleme yapmaları gerekmektedir. Bir düğüm sertifikasını almak istediğinde, parçalı sertifikalar talep ederek kendine ait *k* adet devrelye bir istek gönderir. Birleşme talep edilen devrenin iyi huylu bir devre olduğunu düşündüğünde, ara değer işlevi kullanarak yeni sertifika yayınlamak için daha sonra hedef devre ile birlikte birleşecek parçalırın sertifikalarını yayınlar.

Sertifika yenileme bir yenileme süresi T_{renew} belirterek yürütülür. Bir sertifikayı yenilemek için, bir ağ oluşumu mevcut bulunan geçerli sertifikasını ve tek sekmeli komşularına gelecekteki vade sonu süresini, T(zaman) < (güncel zaman + T_{renew}) olarak yayınlar. Komşu devreler, talebin kabul edildiği ya da reddedildiğini belirlemek için sistem genel anahtarını ve Sertifika İptal Listesini kontrol ederler.

Sertifika iptali, bölüm 2-4-3-1'de net ve gizli mekanizmalar olarak belirtildiği şekilde iki yöntemle yapılır. Gizli mekanizmada, vade bitiş süresi (T_{expire}) veriliş artı yenileme süresinden (T_{renew}) daha az ise iptal edilir. Net sertifika iptal yönteminde, her devre, henüz süresi dolmamış sertifikaları içeren bir Sertifika İptal Listesi tutar. Devre süresi dolmuş sertifikalar için (Certificate Revocation List) CRL'ye periyodik olarak danışır ve eğer gerekliyse onları iptal eder.

Bu yöntemin temel avantajı, herhangi bir merkezi sertifika yetkisi gerektirmemesidir. Ancak, her devrenin doğrulama için en az *k* adet sekmeli komşuları olduğuna güvenir. Bu durum, devrelerin dinamik yapısı dolayısıyla k'nın büyük olduğu durumlarda kullanışlı değildir. Ayrıca, sertifikalar sekme özelliğinden daha fazla özelliğe sahip olan devrelere verilemez. Bu da, ilk olarak *k* devreleri arasında sistem özel anahtarını dağıtmak için bir ön yükleme gerektirir.

2.4.3.3. Kendi kendini Yönetebilen Çok Türlü Sertifikasyon

Wang, Zhu ve Li [21], farklı bir idare alanından olan Tasdik Yetkisinin ağ içinde bir arada var olabileceği özgün bir mekanizma önerisinde bulunur. Aynı zamanda Kong et al [23] tarafından tanıttıkları yönteme benzer *k*-girişi hızlı paylaşım kullanarak yayınlanmış sertifika yetkisi önerirler. Çok türlü Tasdik Yetkilerini ele almak için, güven grafikleri kullanılmıştır. A ve B olarak adlandırılırlar iki devrenin var olduğu varsayıldığında, devre *B*, A'nın halihazırda güvenliği Tasdik Yetkisi tarafından imzalanan B'nin dijital sertifikasına dayanır şekilde özgün olarak doğrulanabilen bir sertifikacı sahip dolayısında *A'nın B'ye* güvenmesi gerektiği belirtilmiştir. Her devre güvenliği Tasdik Yetkisi imzasını taşır.

Bir devrenin bir sertifika elde etmesi gerektiğinde, tek sekmeli komşularından geçerli ortaklarına ait *K* Kimlikleri toplamak ve özel anahtarı düzenlemek zorundadır. Devre *A* başka bir Devre *B*'yi doğrulamak isterse, *B'*ye kendi Tasdik Yetkisini göndererek işlem başlatil. Sonrasında, B, *A*'ya kendi Tasdik Yetkisini gönderir. Daha sonra *A*, müşterek Tasdik Yetkisini kontrol etmek için iki listeyi karşılaştırır, ve eğer yetkisi, *A*, müşterek Tasdik Yetkisi ile onaylanmış şekilde *B*'ye kendi sertifikasını göndermek için işleme başlar. İki devrenin müşterek bir Tasdik Yetkisi olmaması durumunda, o zaman *Dağıtılmış bir Çoklu Sekme Talebi* (DMCR) algoritması aracılığıyla kendilerine ait bir ve iki sekmeyi aramak için işleme başlarlar.

Sertifika yenileme adımları DMCR şemasına benzer. Ancak, bu yönetimde sertifika iptali tartışılmamıştır.

Bu yaklaşımın başlıca avantajları farklı alanlar içindeki Tasdik Yetkileri arasındaki çapraz-sertifikasyon için destek içermesi ve sertifika keşif mekanizmasının *çoklu sekmeler* üzerinde meydana gelimesidir.

2.4.3.4. Güven Gruplaşmaya Dayalı Doğrulama

Ngai et al [24] genel anahtar sertifikasyonunun güvenliğini arttırmak için bir güven ve ağ modeli önerir. Ağ modeli, bazı gruba dayalı algoritmalar ile ağın hiyerarşik organizasyonu ya da gruplaşmasına dayalıdır. Ağın özel Kimlik Numaraları ile gruplara bölündüğünü farz ederler.

Güven modelleri, herhangi bir kullanıcının tasdik makamı olarak hareket edebileceği PGP [29]'a benzeyen güven ağı modeline dayalıdır. Güveni nicel bakımdan 0 ve 1 arası sürekli bir değer olarak tanımlarlar. Her devre ağdaki diğer devreler için bir güven

listesi muhafaza eder. *Doğrudan güven* aynı grup içindeki iki devre arasındaki güven ilişkisi olarak ve *tavsiye edilmiş güven* de farklı gruplardan devreler arasındaki güven olarak tanımlanır. Güven ilişkisi kurmak için, devrelerin, devre hareketini izleyen bir gözlemci gibi bir tespit bileşeni ile donatıldığını farz ederler.

Genel anahtar yönetiminin bir grup içinde var olduğu farz edilmiştir. Ne zaman bir devre başka bir grup içindeki devreyi doğrulamak isterse, adı geçen grup içindeki diğer *tanıtım devreleri* ile iletişime geçer. Güven değerine dayanarak *tanıtım devrelerini* ayırır ve *tanıtım devrelerinin* hedef devreye olan güven değeri ile *tanıtım devrelerinin* güven değerlerini birleştirerek ağırlıklı bir güven değeri hesabı yapar. Daha sonra son güven değeri belirler ve bu değer grup içindeki diğer devreleri değerlendirmek için kullanılır.

Yazarlar, sertifikaların yenilenmesi ve iptali için bir mekanizmadan söz etmezler. Mekanizmanın avantajı, PGP'ye dayalı yöntemlerle karşılaştırıldığında yüksek oranda kötü niyetli devre tespit edebilmesi ve bu devreleri aliğer ile evelerde ayırabilmesidir.

En önemli dezavantajı ise güven değeri ve hesaplamalarının saklanmasının hem bellek hem de zaman alıcı olmasıdır. Ayrıca, devrelerin hareketliliği çeşitli gruplar içinde devrelerin üyeliğinin değiştirilmesine yol açar.

bu bölümde ele alınan mekanizmaların karşılaştırılması tablo 2.2'de özetlenmiştir

Tablo 2.2: Tasarsızlar içinde sertifikaya dayalı doğrulama mekanizması karşılaştırılması.

Gereklilikler	**Kendi kendine organize olan Genel Anahtar Yönetimi – Capkun, vs.**	**Hareketli Geçici Ağ için Sağlam ve Yaygın Güvenlik Desteği Sağlamak – Kong, vs.**	**Hareketli ve Geçici Ağlarda Kendi kendine Yönetilen Çok Türlü Sertifikasyon – Wang, vs.**	**Hareketli Geçici Ağlarda Doğrulamaya Dayalı Güven- ve Gruplaşma - – Ngai, vs.**
R.1. Dağıtılmış doğrulama	Tüm devreler bir Tasdik Yetkisi olarak hareket ettiği için tamamen dağıtılmış bir sertifikasyondur.	Tamamen dağıtılmış ve büyük ağlarda iyi bir şekilde ölçeklenir.	Tamamen dağıtılmış ve büyük ağlarda iyi bir şekilde ölçeklenir.	Tüm devreler bir Tasdik Yetkisi olarak hareket ettiği için dağıtılmış ve kendi kendine organize edilmiştir.
R.2. Kaynak Bilinci	Her devre, yüksek bir destek işlemine maruz kalan iki sertifika deposu muhafaza eder.	Karışık polinominal işlevler kullanan anahtarların oluşum ve dağıtımı kaynağa yoğunlaştırılmış ve zaman alıcıdır.	Her devre sadece kendine ait güvenilir Tasdik Yetkilerini taşır. Dolayısıyla, [21]'de teklif edilen yöntemden daha etkilidir.	Güven tablolarının ve izleme bileşenlerinin koruması belleğe yoğunlaştırılmıştır.
R.3.(a) Oluşum	Kendisi ile imzalanan sertifikalar, ve bu sebeple paylaşılan anahtara dayalı mekanizmalardan daha sağlamlardır.	B ir engel oluşturan k komşularını gerektirir.	K-giriş mekanizmasına benzerdir. [23]	Devreler boyunca, oluşum güven değerlerine dayanır. Devrelerin tanıtımı her zaman doğru olmayabilir.
R.3.(b) Yenilenme	Hiçbir net mekanizma tartışılmamıştır.	Yayın ile aynı	DMCR algoritması aracılığıyla uygulanmıştır	Tartışılmamıştır
R.3.(c) İptal	Net iptal ağlar içindeki uzak devreler arasında gecikmeye sebep olur.	Her devrede saklanan sistem CRL tablosu ve bu sebeple bellek	Tartışılmamıştır.	Tartışılmamıştır
R.4. Çok türkü sertifikasyon	Uygulanmamıştır.	Uygulanmamıştır	Güven grafiği kullanarak uygulanmıştır.	Uygulanmamıştır

2.4.5. Sertifikasyon Mekanizmalarının Değerlendirmesinde Kullanılan Metrikler

Aşağıdaki metrikler, doğrulama mekanizmalarının başarımlarının değerlendirilebilmesinde. kullanılabilecek sayıbal parametreleri içerir [23].

1.Başarılı Sertifikasyon Oranı (μ), başarılı sertifikasyon hizmetleri sayısının (sırasıyla yayınlama, NC_{ISS} ve yenilenme, NC_{REN} dahil) bu hizmetler için olan toplam talep sayısına (sırasıyla $NC_{TOT\text{-}ISS}$ ve $NC_{TOT\text{-}REN}$) olan oranını hesaplar. Mekanizmanın başarılı sertifikasyon hizmetleri sağlamadaki etkinliği ile ilgili fikir verir. μ_{REN} başarılı

sertifikasyon yenileme oranı, μ_{ISS} başarılı sertifikasyon yayınlama oranı ise ilgili değerler aşağıdaki şekilde hesaplanabilir:

$$\mu_{REN} = \frac{NC_{REN}}{NC_{TOT-REN}} \quad \text{(Eş-1)}$$

$$\mu_{ISS} = \frac{NC_{ISS}}{NC_{TOT-ISS}} \quad \text{(Eş-2)}$$

Burada, NC_{REN} ve NC_{ISS} yenilenen ve yayınlanar ilgili toplam sertifika numaralardır ve $NC_{TOT-REN}$ ve $NC_{TOT-ISS}$ sertifika veriliş ve yenilenmesi için ilgili talep sayılarıdır.

2.Yerleşim süresi (st), geçerli sertifikalar verecek ağlar içindeki tüm devreler için başlangıç zamanını ölçer. *st* değeri, tüm devrelerin geçerli sertifikalar verdiği süre ve sertifika süreci verilişinin başlandığındaki süre arasındaki fark olarak hesaplanabilir. Alınan yerleşim süresi, anahtar oluşumu ve dağıtımı vs. için kullanılan kötü niyetli ya da işbirliksiz devreler ve algoritmalar gibi faktörlere bağlı olabilir. Ön doğrulama yöntemleri etkili ise yerleşim süresi daha az olacaktır.

3.Sertifikasyon sıklığı (f_{cert}), sertifikasyon hizmetleri frekansını hesaplar.

$$f_{cert} = \frac{N_{cert}}{T_{int}} \quad \text{(Eş-3)}$$

Burada, N_{cert}, ağ içindeki devrelerle sertifikasyon hizmetlerinin (veriliş/yenilenme) toplam sayısıdır ve T_{int} benzetim süresidir. Bu durum genel şekilde oluşur, çünkü bir devre kendi sertifikasını oluşturmak ya da yenilemek istediği zaman, genel anahtar mekanizması için pahalı hesaplama yapılmak zorundadır. Dağıtılmış ya da kendi kendine organize edilmiş bir mekanizmanın daha düşük bir sertifika oluşum, yenileme ve iptal frekansı ve dolayısıyla daha düşük bir f_{cert}'si olacağını sezgisel olarak öngörmüş bulunmaktayız.

4.Ortalama Sertifikasyon Gecikmesi (acd), benzetim süresi boyunca ortalaması alınmış sertifika hizmet talebi (*CSReq*) ve sertifika hizmet cevabı (*CSRep*) arasındaki süre gecikmesi olarak ölçülmüştür. n devreli bir sistemde acd değeri aşağıdaki gibi hesaplanır.

(Eş-3)

Bu değer, algoritma etkinliğini değerlendirir ve genellikle algoritmaya ait zaman karmaşıklığına bağlıdır.

2.5. YOL ATAMA PROTOKOLLERİNE SALDIRILAR

Tasarsız ağlarda yol atama protokollerine saldırılar 3 sınıfta incelenebilir[4]:

1. Değişiklik Yoluyla Yapılan Saldırılar (Modification)
2. Taklit Edilerek Yapılan Saldırılar (Impersonation)
3. Asılsız Veri Üreterek(Uydurarak) Yapılan Saldırılar (Fabrication)

Bu üç saldırı tipi tasarsız ağlarda taleple tetiklenen protokollerin en yaygınları olan AODV ve DSR göz önüne alınarak anlatılmıştır.

2.5.1. Değişiklik Yoluyla Yapılan Saldırılar (Modification)

Kötü niyetli düğümler denetim mesajlarını değiştirerek veya yönlendirme bilgisindeki değerleri çarpıtarak (hizmet red H.R- DoS) saldırılarına ve ağ trafiğinin yeniden yönlendirilmesine neden olabilir. Aşağıda değiştirme yoluyla AODV ve DSR'ye yapılan birkaç saldırı kısaca anlatılmıştır.

i) **Değiştirilmiş yol sıra numaraları ile yeniden yönlendirme:** AODV ve DSDV (Destination- Sequenced Distance-Vector Routing) gibi protokoller belirli varış noktalarına giden yollara monoton artan sıra numaraları atarlar. Daha yüksek sıra numaralı yol tercih edilir. Bu yüzden AODV'de herhangi bir düğüm, gerçek değerinden daha büyük bir varış sıra numarasına sahip yolun varlığını ilan ederek trafiği kendine çekebilir.

ii) **Değiştirilmiş sekme sayısı ile yeniden yönlendirme:** AODV'de yol bulma mesajındaki sekme sayısı alanı değiştirilerek yeniden yönlendirme saldırısı düzenlenebilir. Yol atama kararı başka diğer metriklerle yapılamadığı zaman AODV sekme sayısı alanını kullanarak en kısa yolu tayin eder. Kötü niyetli

düğümler yol isteğindeki (route request,RREQ) sekme sayısı alanını sıfırlayarak yeni oluşturulacak yolda yer alma şanslarını arttırırlar. Benzer şekilde, sekme sayısı alanını sonsuza götürerek oluşturulacak yolda bulunmamayı da sağlayabilirler. Bu saldırı bir sonraki bölümde anlatılacak olan sızdırma (spoofing) birleştiğinde en tehlikeli saldırı tipi gelebilir.

iii) **Değiştirilmiş kaynak yolu ile hizmeti red saldırısı:** DSR kaynak yol mekanizmasını kullandığı için yolları veri paketlerinde açıkça belirtir. Bu yollarda herhangi bir bütünlük denetimi söz konusu değildir. Basit bir hizmeti red(H.R) saldırısı (DoS) paket başlıklarındaki kaynak yollar değiştirilerek gerçekleştirilebilir. Bu sayedi paketi varış noktasına ulaştırılamaz.

iv) **Tünel saldırısı:** Tasarsız ağlar herhangi iki düğümün komşu olarak konumlanabileceğini varsayar. Tünel saldırısı iki ya da daha çok düğümün varolan veri yolları üzerinde mesaj değiş tokuş etmek üzere birlikte çalışması durumunda oluşabilir. İşbirliği içerisindeki bu kötü niyetli düğümler komşu gibi davranabilir ve iyi niyetli ara düğümlerin yol uzunluğu ölçüsünü düzgün arttırmasını engelleyerek var olan yolların uzunluğunu yanlış gösterebilir.

Kötü niyetli düğümler mevcut çok sekmeli yollar üzerinde tünel kurmak yerine aralarındaki uzun menzilli yönlü telsiz ya da telli bağlantıları da kullanabilir. Böyle bir bağlantı haksız da olsa saldırgan düğümlere başlangıç ve varış arasında en kısa gecikmeli yol üzerinde bulunma üstünlüğünü sağlar. Bu saldırı şekli literatürde solucan deliği saldırısı (wormhole attack) olarak anılır[19]. Ancak kötü niyetli düğümler gerçekten en kısa gecikmeli yol üzerinde ise bu yolun seçilmesi yol atama protokolüne bir saldırı olarak düşünülmeyebilir. Solucan deliği saldırılarına karşı bir savunma mekanizması [20]'de anlatılmıştır.

2.5.2. Taklit Edilerek Yapılan Saldırılar (Impersonation)

Bir düğüm kendisinden çıkan paketlerde IP ya da MAC adresinde değişiklik yapıp kimliğini bilerek yanlış gösterirse ve diğer saldırılarla (değişiklik yoluyla yapılan

saldırı) birleşim içindeyse yalancı kayıt saldırıları (spoofing) meydana gelir. Bu saldırı türünde kötü niyetli düğümün saptanması zordur.

2.5.3. Asılsız Veri Üreterek Yapılan Saldırılar (Fabrication)

Asılsız veri üreterek yapılan saldırıları sahte yol atama mesajları içerir. Özellikle komşu düğüm ile bağlantı kurulamadığını iddia eden yalancı hata mesajının verildiği saldırıların geçersiz olduğunu teyit etmek zor olabilir.

i) **AODV ve DSR'de yol hatalarını çarpıtmak:** AODV ve DSR'de aktif yol üzerindeki varış düğümünün ya da ara düğümlerin hareket etmesi halinde yolda bağlantıda olan düğüm akıştaki diğer aktif düğümlere yol hata mesajı yayımlar. Böylece ilgili yol tüm düğümlerde geçersiz kabul edilir. Sürekli gönderilen ve yol üzerindeki bağlantıyı kopmuş gibi gösteren yol hata mesajları kaynak ve hedef düğümleri arasındaki haberleşmenin önüne geçen bir saldırıdır(DoS).

ii) **DSR'de yol kayıtlarını zehirlemek:** DSR'de bir düğüm kaynak ve hedef arasındaki yolda olmadığı halde kulak misafiri olduğu bir paket başlığındaki yol atama bilgisini kendi yol kayıtlarına ekleyebilir. Saldırgan bu durumdan faydalanır ve paket başlıklarında geçersiz yol içeren paketleri aktararak yol kayıtlarını zehirleyebilir.

2.6. YOL ATAMADA GÜVENLİK GEREKSİNİMLERİ

Güvenli yol atama protokolleri kötü niyetli düğümlerin varlığında kaynaktan hedefe yol bulma fonksiyonunu başarmak için [4]'te 3 ayrı tasarsız ağ ortamı için sıralanan güvenlik gereksinimlerini yerine getirmelidir:

1. Yol işaretleri aldatılmamalıdır.
2. Asılsız yol atama mesajlarının ağa girmesi engellenmelidir.
3. Yol atama mesajları iletimde protokol fonksiyonunun işleyişi dışında değiştirilmemelidir.
4. Zararlı eylemler dahilinde yol atama döngülerinin oluşturulması engellenmelidir.
5. Yeniden yönlendirmeye izin verilmemeli, en kısa yollar korunmalıdır.

Bu sistemde, *açık* ortamda tüm düğümler yetkili olarak kabul edilmiştir. Otoyolda araba kullanan ya da kentte yürüyen bu bir kişi ortamda yetkili bir düğüm olarak düşünülebilir.
Yetkili olmayan düğümlerin yol bulma ve hesaplamadan men edilmesi ile *kontrollü açık* ortam gereksinimi de karşılanmış olur. Ancak bu gereksinim yetkili düğümlerin de kötü niyetli davranabileceği gerçeğini ortadan kaldırmaz. Kontrollü açık ortamlarda açık anahtarların, oturum anahtarlarının ya da sertifikaların önceden belirlenmesi olanağı olduğu kabul edilir. Bu ortamdaki gezgin düğümlerin ortak bağlamda ve yakın coğrafyada olması beklenir. Toplantı katılımcıları, kampüsteki öğrenciler bu ortamda düşünülebilir.

Kontrollü düşman ortamında ise yukarıdaki gereksinimlerin yanı sıra ağ topolojisinin ne düşmana ne de yetkili kabul edilen düğümlere açık bırakılmaması istenir. Bu ortam savaş meydanındaki askeri düğümler ya da felaket bölgesindeki kurtarma ekibi için oluşturulabilir. Böyle bir ortamda düğümler müşterek bir kaynak tarafından yerleştirilebilir. Böylece güvenlik parametreleri önceden belirlenebilir. Bu ortamdaki ayırt edici güvenlik tehlikesi donanımın fiziksel olarak ele geçirilmesi ve yönetiminin ele alınması ile dost düğüm taklidi yapılmasıdır. Bu yüzden yol atama protokolü mesajlarında düğüm konumunun duyurulması istenmez.

2.7. ULAŞIM KATMANINDA GÜVENLİK

Tasarsız ağlarda düğümlerin asıllanması ve yol atama bilgilerinin bütünlüğünün korunarak onaylanması güvenliğin temel yapıtaşlarıdır. Her iki yapıtaşı da uygun anahtarların kullanılmasında bir *anahtar yönetim mekanizması*nın varlığını gerektirir. Önerilen güvenli yol atama protokolleri için anahtar yönetimi yaklaşımı iki kategoride toplanabilir.

2.7.1. Simetrik Gizli Anahtarların Düzenlenmesi

Gizli anahtarlar haberleşen düğümler arasında oturum anahtarı oluşturmada kullanılabilir. Anahtar değerlerini içinde barındıran adanmış sunucunun bir yapı olduğu

kurulabilirse, oturum anahtarlarının Kerberos gibi bir anahtar dağıtım protokolü ile otomatik dağıtımı söz konusu olabilir.

2.7.2. Açık Anahtar Tabanlı Düzen

Her düğüm RSA gibi asimetrik algoritma tabanlı bir gizli-açık anahtar çiftine sahiptir. Bu anahtar çifti ile asıllama ve bütünlüğü koruma fonksiyonlarını yerine getirir.

SRP(Secure Routing Protocol), paylaşılan gizli anahtar tabanlı bir güvenlik birliği düzeni üzerine çalışır. Önerilen diğer yol atama protokollerinin çoğu, örneğin ARIADNE, asıllamayı sağlamak için güvenilen üçüncü bir otoritenin(TTP, trusted third party) açık anahtar sertifikalarını yayınladığı açık anahtar düzeneğine güvenmektedir.

Aşağıda tasarsız ağlarda açık anahtar altyapısının karışıklığını ele alan iki yaklaşım anlatılmıştır. İlki PGP'ye benzer kendi kendine düzenlenen açık anahtar yönetimidir. İkincisi ise polinom şeklindeki gizli bilgi paylaşımına dayanan bir açık anahtar sertifika mekanizmasıdır.

2.7.2.1. PGP'ye Dayalı, Kendi Kendine Düzenlenen Açık Anahtar Yönetimi

[20]'de açık anahtar tabanlı yapı ele alınarak kendiliğinden düzenlenen açık anahtar yönetim sistemi önerilmiştir. Bu yöntem kendiliğinden yapılanan tasarsız ağ kavramının doğası ile örtüşmektedir. Sabit bir sunucuya ya da güvenilir bir otoriteye ihtiyaç duyulmadan düğümlerin kendi açık anahtarlarını üretmeleri, saklamaları, dağıtmaları ve imha etmeleri esas alınmıştır. Bu yöntemde herhangi bir şekilde bir düğüm kümesinin belirli görevleri olması beklenmez. Her düğüm eşit şartlardadır. Bu yaklaşım merkezi bir denetim olmadan düğümlerin ağa girip çıkabildiği açık ortamlar için geliştirilmiştir.

PGP'de de olduğu gibi sertifika mekanizması, kullanıcıların kişisel bilgilerine dayanır. Ancak PGP'de sunuculardaki sertifika klasörlerinde tutulup dağıtılan sertifikalar bu modelde kullanıcıların kendisi tarafından gerçeklenir. Kullanıcılar sertifikalarını yerel kullanıcı depolarında saklarlar. İki kullanıcının birbirlerinin açık anahtarlarını onaylaması gerektiğinde sertifika depolarını birleştirip bu birleşim içinde uygun sertifikaları bulmaya çalışırlar.

Bu yaklaşımın başarısı büyük ölçüde yerel sertifika deposu yapısından ve sertifika grafiklerinin karakteristiğinden kaynaklanmaktadır.

2.7.2.2. Polinom Şeklindeki Gizli Bilgi Paylaşımına Dayalı Asıllama

Ağın erişimini iyi niyetli düğümlerle sınırlamak bütün düğümlerin işbirliği içinde olduğu güvenli tasarsız ağlarda kritik bir işlemdir. Bunu sağlamak için [22]'de önerilen yaklaşımda düğümlerin ağa erişimi için geçerli bir bileti olması beklenmektedir. Gizli sayısal imza anahtarı, açık anahtar sertifikalarını üretmeleri için polinom şeklindeki gizli bilgi paylaşımına dayalı bir grup imzası mekanizması ile çok sayıda düğüme dağıtılır. Böylece yayınlama, yenileme ve imha etme sertifika hizmetleri de bu düğümlere dağıtılmış olur. Hiçbir düğüm tamamen güvenilir değildir ve erişimi tekeline alamaz. Birden fazla düğüm(komşu düğümler) birlikte başka bir düğümün davranışını gözetler, biletini onaylar ya da imha eder (localized trust model). Düğümlerin taşıdığı sertifikalar ortak sertifika anahtarı (SK) ile imzalanmıştır. SK'nın bir şekilde her düğüme iletilmesi gerekir. Bu işlemi başlatmak için en az ilk k düğüme bu anahtarın verilmesi gerekir. Bu düğümler işbirliği içinde anahtarı diğer düğümlere yayarlar.

Düğümler yer değiştirdiğinde yeni komşuları ile güven ilişkisine girme isteğinde bulunur.

Kötü niyetli düğümler için tüm ağda *suçlama* duyurusu yapıldığından kötü niyetli düğümler sürekli yer değiştirerek faaliyetlerine devam edemezler.

3. MALZEME VE YÖNTEM

3.1. TASARSIZ TALEP ESASLI UZAKLIK VEKTÖRÜ PROTOKOLÜ (AODV)

Tasarsız Talep Esaslı Uzaklık Vektörü (AODV) yönlendirme algoritması, tasarsız mobil ağlar için tasarlanmış olan bir yönlendirme protokolüdür. Bu protokol dinamik bağlantı durumlarına, düşük işlem ve bellek proseslerine çabuk adaptasyon sağlar. Klasik uzaklık vektörü protokolleri ile ilişkili döngü problemlerini çözmek için (sonsuza kadar sayma problemi gibi) hedef sıra numaraları kullanır.

3..1. AODV'nin Özellikleri

Tasarsız Talep Esaslı Uzaklık Vektörü (AODV) yönlendirme algoritması, talep esaslı bir algoritmadır, yani sadece kaynak düğümler tarafından ihtiyaç duyulması durumunda düğümler arasında rotalar inşa eder ve bu rotalar sadece gerekli oldukları müddetçe tutulur. AODV'de, kopuk bağlantıları onarırken bile hiçbir zaman döngü oluşumu söz konusu değildir. Komşu topolojide bir değişim hakkında bilgi sahibi olduğu her an sıralı olarak artan bir sıra numarası oluşturur. Bu sıra numarası, her rota araması yapıldığında en güncel rotanın seçilmesini sağlar. AODV teke gönderimli, çoğa gönderimli ve geniş gönderimli haberleşme yeteneğine sahiptir. Ayrıca AODV, çok gönderimli grup üyelerini birbirine bağlayan ağaçlar oluşturur. Bu ağaçlar grup üyelerinden ve üyeleri birbirine bağlamak için gerekli olan düğümlerden meydana gelmektedir. Son olarak temel algoritmaya yapılan düzenli iyileştirmelerin (mesela QoS uygulamaları için, istemci-sunucu aramaları için veya asimetrik yönlendirme yollarının kullanımı için) hem tekli hem de çoğa gönderim tabanlı veri iletimine fayda sağlaması beklenmektedir.

AODV halihazırda komşu düğümler arasında sadece simetrik bağlantılar kullanmaktadır. İlgili yönlendirme bilgilerini kaydetmek için hem teke gönderim hem de çoğa gönderim yönlendirme tabloları kullanılır. Bu tablolar, sabit hareket halindeki düğümler için bile hem teke gönderim hem de çoğa gönderim rotalarını tutabilir. Ayrıca özel bir rota hata mesajı kullanımı vasıtasıyla geçersiz rotaların çabuk şekilde silinmesine olanak verir. AODV, çabuk ve dakik bir şekilde aktif rotaları etkileyen topolojik değişikliklere cevap verir. Rotaları, yönlendirme kontrol mesajlarından doğan sadece ufak miktarda bir protokol yükü ile ve herhangi ekstra bir ağ protokol yükü

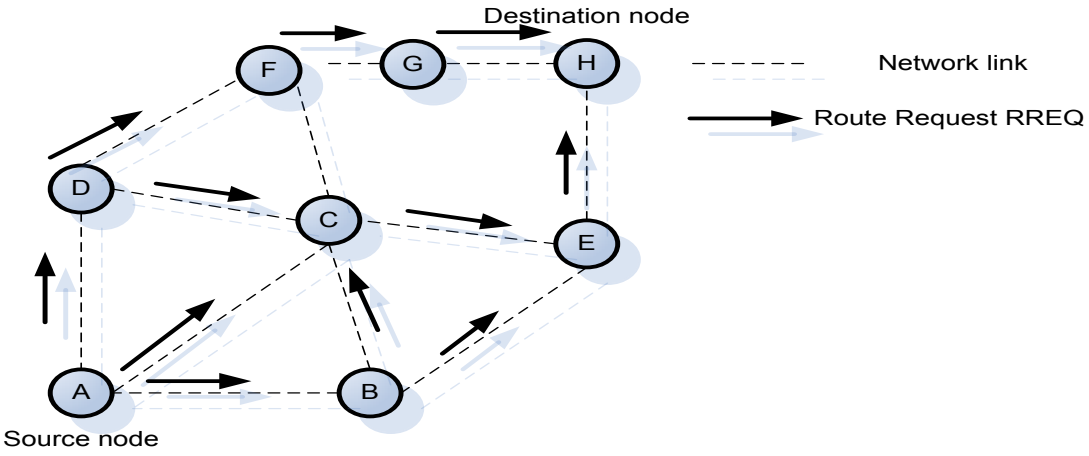
olmadan inşa eder. Sonuç olarak AODV veri paketleri üzerine herhangi ekstra bir protokol yükü yüklemez çünkü kaynak yönlendirmesi kullanmaz.

3.1.2. Unicast (Teke Gönderim) Rota Oluşturma

AODV ile rota bulma tamamen talep esaslıdır ve bir rota talep/rota cevap arama döngüsünü takip eder. Talepler bir Rota Talep (RREQ) mesajı kullanılarak gönderilir. Bir Rota Cevap (RREP) mesajı ile, bir rota oluşum bilgisi geri gönderilir.

3.1.3. Rota Arama

Bir kaynak düğüm belli bir hedef düğüme veri paketi göndermek istediğinde ve bu hedef düğüm için geçerli bir rotası olmadığında bir rota arama prosesi başlatır. Böyle bir işlemi başlatmak için kaynak düğüm bir RREQ paketi oluşturur. Kaynak düğümün IP adresine, mevcut sıra numarasına ve geniş gönderim ID'sine ilave olarak RREQ aynı zamanda hedef için kaynak düğümün bildiği en güncel sıra numarasını da içerir. RREQ oluşturulduktan sonra kaynak düğüm bu paketi komşusuna gönderir. Buradan talep komşulara aktarılır ve bu işlem, hedef düğüme veya "hedefe en güncel rota bilgisine" sahip ara düğüme ulaşıncaya kadar devam eder ve sonra bir zamanlayıcı ayarlanacak beklemeye alınır.



Şekil 3.1: Ağ içerisinde yayılma veya RREQ.

Bu RREQ'u işlemek için her düğüm, kaynak düğüm için rota tablosu içerisinde, kaynak düğümün IP adresini ve sıra numarasını ve ayrıca kaynak düğümün IP adresine kadar olan sekme sayısını ve sıra numarasını ve aynı zamanda RREQ bilgisinin alınmış

olduğu komşu düğümün IP adresini içeren bir ters rota kaydı oluşturur. Şekil 3.1, RREQ'un ağ genelinde yayılımını ve ayrıca her ağ düğümünde ters rota kayıtlarının oluşumunu göstermektedir. Her rota kaydı ile ilişkili olarak, belirlenen periyotta kullanılmaması durumunda kaydın silinmesini sağlayacak bir rota zamanlayıcısı mevcuttur.

RREQ'u alan düğüm, hedef düğüm olması durumunda veya RREQ içerisinde yer alan sıra numarasına eşit veya bu numaradan büyük olan sıra numarasına sahip olması durumunda bir rota cevabı (RREP) gönderebilir. Eğer durum böyleyse kaynağa RREP bilgisini teke gönderim yöntemiyle geri iletir. Aksi takdirde RREQ bilgisini komşu hücrelere aktarmaya devam eder.

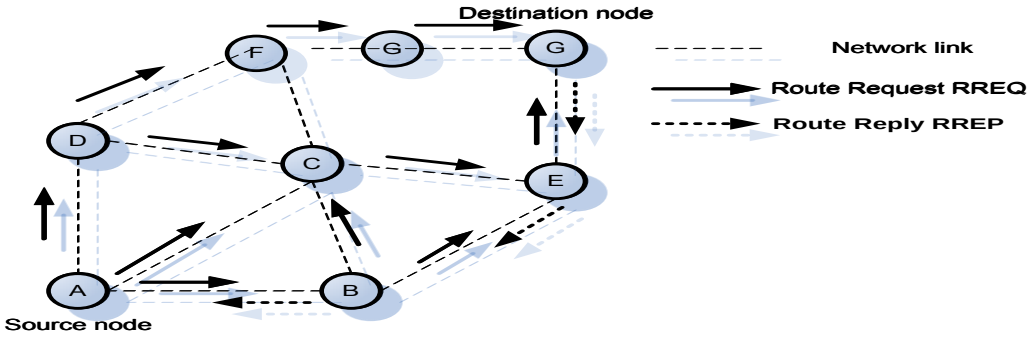
3.1.4. Geniş Alanda Arama (Expanding Ring Search)

Düğüm bazı yeni hedefler için bir rota arama işlemi başlattığı her an ağ geneline bir RREQ'u yayınlamak zorundadır. Ufak bir ağ için bu sel algoritmasının etkisi az olabilir fakat büyük bir ağ için bu etki yüksek ve zarar verici olabilir. Tüm ağ geneline yayınlanan RREQ'ları kontrol etmek için kaynak düğüm bir "geniş alanda arama tekniği" kullanmak zorundadır. Bu teknik sayesinde, hedefe giden bir rota bulunamazsa ağ içerisinde kapsamı gittikçe artan alanlarda arama yapılır. Bu arama tekniğini kullanmak için kaynak düğüm, RREQ'un Yaşam Süresini (TTL) başlangıç ttl_start değerine getirir. Eğer arama süresi içerisinde herhangi bir cevap alınamazsa değeri artış değeri kadar arttırılarak bir sonraki RREQ yayınlanır. TTL değerini bu şekilde arttırma işlemi bir eşik değere ulaşılana kadar devam eder.

3.1.5. İleri Yol Yapılandırması (Forward Path Setup)

Sonuç olarak RREQ, istenen hedefe giden mevcut rotalardan birini barındıran bir düğüme (muhtemelen hedefin kendisi) ulaşacaktır. RREQ mesajını alan düğüm ilk olarak RREQ'un çift yönlü bir bağlantı üzerinden alınıp alınmadığını kontrol eder. Eğer ara bir düğüm istenen hedef için rota kaydına sahipse, istenen hedef için kendi rota kaydı içerisinde sıra numarasını RREQ içerisindeki hedef sıra numarasıyla karşılaştırmak suretiyle bu rotanın güncel olup olmadığını tespit eder. Eğer hedef sıra numarası ara düğümde kayıtlı olan sayıdan büyükse, ara düğüm RREQ cevap vermek için kendi kayıtlı rotasını kullanmamalıdır. Bunun yerine yapması gereken şey RREQ yeniden ağ geneline yayınlamaktır. Ara düğüm, ancak RREQ'a içerisinde kayıtlı sıra numarasına eşit veya bu numaradan büyük olan bir sıra numarası olan bir rotaya sahipse

cevap verebilir. Eğer istenen hedef için bir rotaya sahipse ve eğer RREQ'u daha önceden işleme tabi tutulmadıysa, düğüm, RREQ mesajını almış olduğu komşusuna teke gönderim şekliyle bir rota cevap paketi (RREP) gönderir. İstenen hedefe rota temin edebilecek bir düğüme bir yayın paketi ulaşana kadar RREQ kaynağına bir ters yol oluşturulmuş olur. RREP, kaynağa doğru geri yol aldıkça yol boyu her düğüm RREP'in geldiği düğüme ileri yönlü bir işaretçi koyar, kaynağa ve hedefe olan rota kayıtları için zaman aşımı bilgilerini günceller ve istenen hedef için en son hedef sıra numarasını kaydeder. Şekil 3.2, RREP'in hedeften kaynağa doğru ilerlemesi esnasında söz konusu olan ileri yol yapılandırmasını göstermektedir. RREP tarafından tespit edilen yol boyunca yer almayan düğümler, ACTIVE_ROUTE_TIMEOUT süresi sonrasında (3000/msec) zaman aşımına uğrayacak ve silinecektir.



Şekil 3.2: Kaynaktan Hedefe Rota Tayini.

Eğer düğüm birden fazla RREP alırsa ilkini ve bir önceki RREP'den daha büyük veya bu RREP ile aynı hedef sıra numarasına ve daha küçük sekme sayısına sahip olanları ağa yayar.

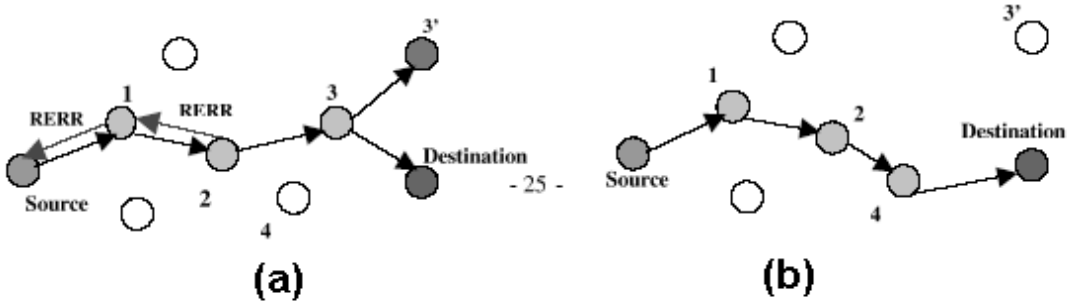
3.1.6. Rota Tablosu Yönetimi

AODV, her rota tablosu kaydı için şu parametrelerin kaydını tutmak zorundadır.

- Hedef IP Adresi: Hedef düğüm için IP adresi
- Hedef sıra numarası: Bu hedef için sıra numarası
- Sekme sayısı: Hedefe doğru sekme sayısı
- Sonraki sekme: Bu rota kaydı için hedefe paket sevketmek üzere tahsis edilmiş olan komşu düğüm
- Yaşam süresi: rotanın geçerli olarak kabul edildiği süre
- Aktif komşu listesi: Bu rota kaydını aktif olarak kullanan komşu düğümler

3.1.7. Rotanın Tutulması

Belirli bir kaynak/hedef çifti için bir rota bulunduğunda bu rota, kaynak düğüm tarafından ihtiyaç duyulduğu müddetçe tutulur. Tasarsız ağ içerisinde düğümlerin hareketi sadece bu düğümleri içeren rotaları etkiler. Bu yol aktif yol olarak isimlendirilir. Aktif bir yol boyunca söz konusu olmayan hareket herhangi bir protokol işlemini tetiklemez. Eğer kaynak düğüm aktif bir oturum esnasında hareket ederse, hedefe yeni rota tespit etmek amacıyla rota arama işlemini yeniden başlatabilir. Hedef düğüm veya herhangi bir ara düğüm hareket ederse etkilenen kaynak düğüme bir Rota Hatası (RERR) mesajı gönderilir. Bu RERR, kesintinin yukarı yönündeki düğüm tarafından tetiklenir (yani kaynak düğümlere yakın olan). Bu mesaj, bağlantı kaybından ötürü o an itibariyle erişilemeyen durumdaki her hedefi listeler. Eğer kesintinin yukarı yönündeki düğüm, bağlantı kaybından ötürü bir veya daha fazla erişilemeyen düğüme sahipse veya hedef için öncü düğüm olarak listelenmiş bir veya birden fazla düğüme sahipse (bir veya daha fazla düğüm hedefe ulaşmak için bu öncü düğüm üzerinden geçiyor, beli RERR mesajını bu komşu düğümlere yayınlar. Komşular RERR mesajını aldığında, hedefe olan mesafeyi sonsuz yapmak suretiyle hedefe olan rotalarını geçersiz olarak işaretler ve sırasıyla RERR mesajını öncü düğümlerine yayar (rota tablolarında hedefler için bu gibi düğümler listelenmişse). Kaynak düğüm RERR mesajını aldığında, rotaya halen ihtiyaç varsa rota arama prosedürünü yeniden başlatabilir.



Şekil 3.3: Rotanın tutulması.

Şekil 3.3, rota tutma prosedürünü göstermektedir. Şekil 3.3(a)'da kaynaktan hedefe giden orijinal yol 1, 2 ve 3 no'lu düğümler üzerinden geçmektedir. Düğüm 3 daha sonra 4 konumuna hareket ederek düğüm 2 ile aradaki irtibatı kesintiye uğratır. Düğüm 2 bu kesintiyi algılar ve düğüm 1'e RERR mesajı gönderir. Düğüm 1 bu rotayı geçersiz olarak işaretler ve RERR'yi kaynağa yönlendirir. Kaynak düğüm RERR'yi aldığında halen rotaya ihtiyacı olup olmadığını tespit eder ve gerekiyorsa rota arama prosedürünü

yeniden başlatır. Şekil 3.9 (b), düğüm 4 aracılığıyla bulunmuş olan yeni rotayı göstermektedir.

3.1.8. Yerel Bağlantısallık Yönetimi

Komşu bilgiler, komşu düğümler tarafından gönderilen paket yayınlarından elde edilir. Bir düğüm komşu bir düğümden veri yayını aldığında yerel bağlantısallık bilgisini güncelleyerek bu komşu düğümü kapsamına alır. Eğer tablo içerisinde halihazırda bu komşu için herhangi bir kayıt yoksa düğüm bu kaydı oluşturur. Eğer bir düğüm aşağı yönündeki aktif komşularının tümüne hello_ süre aralığında herhangi bir paket göndermezse komşularına, kimliğini ve mevcut sıra numarasını içeren bir Hello mesajı (talep edilmemiş özel bir RREP) yayınlar. Bu mesaj, değeri 1 olan bir TTL içerdiğinden düğümün komşuları dışına yeniden yayınlanamaz. Çeşitli Hello_ mesajlarının periyodik iletimi tarafından tanımlanan süre içerisinde bir komşudan herhangi bir iletim alınamaması, yerel bağlantısallığın değiştiğine ve bu komşu için rota bilgilerinin güncellenmesi gerektiğine işaret eder.

3.1.9. Sistemi Yeniden Açma Sonrası İşlemler

Sistem yeniden açıldığında, bir düğüm bir önceki sıra numarasını ve aynı zamanda diğer hedefler için bilinen son sıra numaralarını kaybetmiş olacaktır. Komşu düğümler bu düğümü sonraki aktif sekme olarak kullanabileceğinden yönlendirme döngüleri yaratabilir. Bu ihtimali ortadan kaldırmak için sistem yeniden açıldığında her düğüm delete_ süresi kadar bekler ve bu esnada herhangi bir yönlendirme paketine cevap vermez. Fakat bir veri paketi aldığında RERR mesajı yayınlar ve mevcut zaman + delete_ süresi sonrasında sona ermek üzere bekleyen zamanlayıcıyı (yaşam süresi) sıfırlar.

3.1.10.Yayın

AODV, paket yayınlarının iletim davranışını belirler. Bir düğüm paket yayını üretmek istediğinde yayın paketini bilinen yayın adresi olan 255.255.255.255'ye gönderir.

Düğüm 255.255.255.255 adresine gönderilmiş bir paket yayını aldığında kaynak IP adresini, IP boşluk değerini ve paketin IP başlığının etkisini işaretler. Daha sonra paketin alınıp alınmadığını ve buna bağlı olarak yeniden iletilip iletilmediğini tespit etmek üzere yayın listesi kayıtlarını kontrol eder. Eğer karşılık gelen böyle bir kayıt yoksa düğüm ilgili yayın paketini işler ve yeniden iletir. Aksi durumda paketi reddeder.

3.1.11. Optimizasyon İşlemeleri:

Çeşitli simülasyonlar ile AODV çeşitli ağ senaryoları içerisinde mükemmel performans göstermiştir. Fakat ekstra servisler sağlaması ve çok yönlülük kazanması yönünde halen yapılacak iyileştirmeler ve geliştirmeler vardır. Aşağıdaki bölümler, AODV'yi optimize etmeye yönelik yöntemleri açıklamaktadır.

3.1.11.1. Servis Kalitesi

AODV, belirli servis kalitesi parametreleri talep etmek için kullanılabilir. Bunlar bilhassa bir RREQ'e ilave edilebilen Maksimum Gecikme ve Minimum Band Genişliği gibi parametrelerdir. Bu uzantılara sahip bir RREQ'a cevap vermek için düğüm, belirtilmiş servis kalitesi sınırlamalarını sağlamalıdır. Eğer böyle bir rota oluşturulduktan sonra yol boyunca yer alan herhangi bir düğüm talep edilen servis kalitesini artık koruyamadığını algılarsa, bu talepte ilk olarak bulunmuş olan düğüme ICMP QOS_LOST mesajı göndermek zorundadır.

Minimum Band Genişliği uzantısı için gerekli olan herhangi bir ekstra işlem gerekmez çünkü ara düğümler, bu uzantıya sahip olan RREQ mesajını yeniden yayınlamaktadır. Maksimum Gecikme uzantısı için her ara düğüm, sonraki veri paketlerinin işlenmesine gecikme ilave etmek için kendi karakteristik değerini çıkarmak zorundadır. Bu karakteristik değer ölçülmüş işlem zamanına ve aynı zamanda zamana göre değişen kuyruk uzunluklarına bağlıdır. Gelen veri paketleri için kuyruk uzunluklarının son derece değişken olması beklenir bu yüzden önceden belirlenmiş konservatif bir metrik değer hazırlanır. Ayrıca talepte bulunan düğümden gelen paketler için kuyruk gecikmelerini azaltmak amacıya böyle bir sabit gecikme zamanı öneren bir düğüme ihtiyaç duyulabilir. Aynı IP kaynak düğümünden gelen çeşitli tiplerdeki akışlar arasında gönderici düğümün ayrım yapmasını sağlamak için ekstra çalışma gerekli olabilir.

3.1.11.2. Alt Ağ Yönlendirmesi

Alt ağ yönlendirmesinin yapıldığı durumlarda grup içerisindeki düğümlerden herhangi birine olan rota, hemen hemen bir sekme ilerideki veya gerideki herhangi diğer bir düğüme giden rota kadar iyidir. Bu yüzden tüm düğümlere giden rota bilgiler tek bir rota tablosu kaydı ile özetlenebilir ve rota kümelemesi yapılabilir.

AODV ile çalışmak için alt ağa giden rotalara, tıpkı çoğa gönderim gruplarına bir sıra numarası atanır gibi bir hedef sıra numarası atanmalıdır. Alt ağ için gereken tüm işlem alt ağ üzerindeki düğümlerden birinin sıra numarası oluşturma ve yönetme

sorumluluğunu üzerine almasıdır. Eğer alt ağda bir yöneltici (router) varsa bu düğüm bu amaçla en mantıklı tercih olacaktır. Eğer yoksa başka bir düğüme bu fonksiyon ve aynı zamanda alt ağ üzerindeki diğer düğümler için trafiği yönlendirme fonksiyonu atanmalıdır. Sıra numarasını yöneten düğüm alt ağ öncüsü olarak isimlendirilir ve bu tüm alt ağ düğümleri için yöneltici olarak kabul edilmelidir.

Alt ağ üzerinde RREQ mesajını alan düğümler bu mesajları alt ağ öncüsüne yönlendirmek zorundadır. Alt ağ öncüsü, alt ağ düğümlerine, ağ içerisindeki herhangi diğer bir düğüme yaptığı şekilde bir ters rota yaratır. Alt ağ üzerindeki herhangi bir düğüm üzerinden akan RREP mesajları alt ağ öncüsü vasıtasıyla kaynağa geri gönderilmelidir. Alt ağ içerisinde tüm rota oluşumları talep esaslı olsa da herhangi periyodik bir rota ilanına gerek yoktur

3.2. DİNAMİK KAYNAK YÖNLENDİRMESİ (DSR)

3.2.1. Protokol Tanımı

Dinamik Kaynak Yönlendirmesi (DSR) de yine talep esaslı protokoller sınıfına aittir ve düğümün çoklu ağ sekmeleri boyunca herhangi bir hedefe giden rotayı dinamik olarak keşfedebilmesine olanak verir. Kaynak yönlendirmesi, her paketin kendi üstbilgisi içerisinde paketin geçmesi gereken eksiksiz düğümler listesini taşıdığı anlamına gelir. DSR periyodik yönlendirme mesajları kullanmaz (mesela yöneltici ilanları) ve böylece ağ band genişliği üzerindeki protokol yükünü azaltır, elektrik enerjisinden tasarruf sağlar ve tasarsız ağ genelinde geniş kapsamlı yönlendirme güncellemelerinin yapılmasını ortadan kaldırır. Bunun yerine DSR, MAC katmanından gelen desteğe bağımlıdır (MAC katmanı yönlendirme protokolünü bağlantı hatası hakkında bilgilendirmelidir). DSR'deki iki ana çalışma modeli rota araması ve rota muhafazasıdır.

3.2.2. Rota Arama

Rota arama, D hedefine paket göndermek isteyen S düğümünün D hedefine kaynak rotayı elde etmesi için gereken. Bir rota oluşturmak için S kaynağı, kendi kablosuz iletim alanı içerisindeki host makinalar tarafından alınabilecek olan ve benzersiz bir talep ID'sine sahip bir Rota Talep (RREQ) paketi yayınlar. Bu talep mesajı hedefe veya hedefe giden rota bilgisine sahip olan düğüme ulaştığında bu noktadan kaynağa yol

bilgisini ihtiva eden bir Rota Cevap (RREP) mesajı geri gönderilir. Her düğümde tutulan "rota arabelleği", bir ağ arama prosedürü tarafından yaratılan protokol yükünü azaltmak amacıyla düğümün zaman içerisinde öğrendiği rotayı ve protokol yükünü kaydeder.

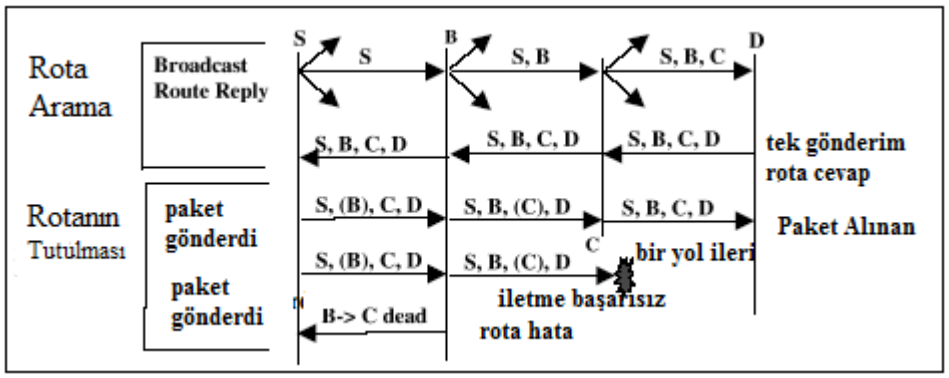
Düğüm bir rota talep paketi aldığında bu talebi aşağıdaki adımlara göre işleme tabi tutar:

- Eğer bu rota talebi için çiftli parametre (talepte bulunanın adresi, talep ID) host makinanın son görülen talepler listesinde bulunuyorsa o zaman rota talep paketi dikkate alınmaz ve üzerinde herhangi bir işlem yapılmaz.
- Eğer host makinanın adresi talep içerisindeki rota kaydında bulunuyorsa o halde rota talep paketi dikkate alınmaz ve üzerinde herhangi bir işlem yapılmaz.
- Eğer düğüm istenen hedefse, rota kaydı tamamdır, rota cevabı verilir.
- Eğer düğüm rota talep paketi içerisindeki rota kaydına eklenir ve mesaj yeniden yayınlanır.

3.2.3. Rotanın Tutulması

Rotanın tutulması, bir S paket göndericisinin, ağ topolojisinde bir değişimin meydana geldiğini ve buna bağlı olarak D hedefine olan rotasını artık kullanamadığını algıladığı bir mekanizmadır. Bu durum, bir kaynak rotası içerisinde yer alan bir host makinanın kablosuz iletim kapsamının dışına çıkması veya kapatılarak rotayı kullanılamaz hale getirmesi durumunda meydana gelebilir. Kopuk bağlantı ya aktif olarak onayları gözlemek suretiyle ya da pasif çalışma durumunda bir paketin komşu bir düğüm tarafından yönlendirildiğini rasgele farkederek algılanır.

Rota tutma mekanizması, kullanımda olan bir rotaya ilişkin bir problem tetkik ettiğinde S kaynak düğümüne bir Rota Hata paketi (RERR) gönderilr. Bu RERR paketi ulaştığında host makinanın rota arabelleğinden hatalı sekme temizlenir ve bu sekmeyi ihtiva eden tüm rotalar bu noktada kırpılır. Şekil 3.4 temel DSR faaliyetini göstermektedir.



Şekil 3.4: Temel DSR Faaliyeti.

3.2.4. Optimizasyonlar

Performansı arttırmak ve protokol yükünü azaltmak için DSR'de birkaç optimizasyon yapılabilir. Bunlardan bazıları şöyledir:

i) **Rota Arabelleğinin Tam Kullanımı**: Bir host makina rotası içerisindeki veri herhangi bir formatta saklanabilir fakat aktif rotalar, arabelliklerinde bir rota ağacı oluşturur ve bu ağaç ilgili host makinadan ağ içerisindeki diğer host makinalara giden rotaları listeler. Bir rota, yeni bir rota hakkında bilgi sahibi olduğu her an rota arabelleğine yeni kayıtlar ekleyebilir. Bir host, paket içerisindeki rota üzerinde bir ara sekme olarak bir veri paketi gönderirse bu host paket içerisindeki tüm rotayı gözleyecektir. Eğer başka bir hosttan kendisine bir rota cevap paketi gönderilirse gelen bu cevap içerisinde yer alan rota kaydından rota bilgilerini kendi rota arabelleğine ekleyebilir. Tüm kablosuz ağ iletimleri doğaları gereği yayın yaptığından, bir host, ağ arayüzünü her paketi alma moduna konfigüre edebilir ve kulak misafiri olduğu herhangi bir rota cevap paketinden veya herhangi bir veriden gelen rota bilgilerini kendi rota arabelleğine ekleyebilir.

Bir host, başka bir hosttan alınmış olan bir rota talep paketinin ağa yayılmasını önlemek için kendi rota arabelleğini kullanabilir. Bu durum şekil 3.5'de göstermektedir mesela eğer F mobil host D mobil hosta bir paket göndermek zorundaysa bir rota arama prosedürü başlatacak ve bir rota talep paketi yayınlayacaktır. Eğer bu yayın A'ya ulaşırsa A hemen D hedefine giden tam rotayı ihtiva eden ve ayrıca A, B, C ve D sekmelerinin sırasını barındıran rota cevap paketini F'ye gönderebilir.

Fakat birden fazla mobil host ilgili düğümün rota talep paketini aldığında ve tümü kendi rota arabellekleri içerisinde bulunan rotalara dayalı olarak cevap gönderdiklerinde problem ortaya çıkabilir. Bu örnekte ikiden fazla mobil host olduğunda, yayını alan mobil hostların aynı anda vereceği cevaplar bu cevapların tümü veya bazıları arasında paket çarpışmasına ve kablosuz ağda lokal sıkışıklığa yol açabilir.

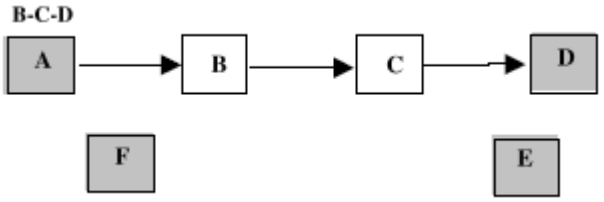
Bu problemi ortadan kaldırmak için her host, aşağıdaki denklemde belirtilen oranda bir gecikme uygular.

gecikme süresi d= H* (h-1+r) (Eş -1)

Burada h: bu hostun cevabı içerisinde belirtilecek rota için söz konusu olan ağ sekmesinin sayısı, r: 0 ve 1 arası rasgele bir sayı, ve H: arabellekten cevap vermeden önce sekme başına öngörülmüş olan ufak bir sabit gecikmedir. Hedefe giden rotadaki sekme sayısına orantılı olan bir gecikme süresinin kullanılması, kaynağın ilk olarak en kısa rotayı alma ihtimalini arttırmaktadır.

Hostlar rota taleplerine kendi arabelleklerinden cevap verirken gündeme gelen bir diğer problem de, rota cevap paketi içerisinde gönderilen rotada döngü oluşmasıdır. Bu problemi gidermek için, eğer bir host bir rota talebi alırsa ve bu talebin hedefi durumunda değilse fakat kendi arabelleğinden cevap verebilme durumuna sahipse ve verilecek cevaptaki rota bir döngü içerecekse bu talebi reddedebilir.

Sonuç olarak kaynak düğümler, TTL değerini sıfır sekmeye ayarlamak suretiyle ağa yayılmayan rota talepleri üretebilir. Böylece sadece komşuların bu mesajı alması sağlanır ve onların da mesajı daha ileri yayınlamaları önlenmiş olur. Bu teknik, kaynak düğümün, hedefe giden rotaya yönelik olarak komşularının arabelleğini sorgulamasına olanak verir. Eğer verilen bir zaman aralığı içerisinde bir cevap alınamazsa düzenli bir rota talep mesajı gönderilir. Bu teknik protokol yükünü azaltma, enerji etkinliğini arttırma ve sınırlı kapasiteli bağlantılar üzerinde tıkanmayı azaltma potansiyeline sahiptir.

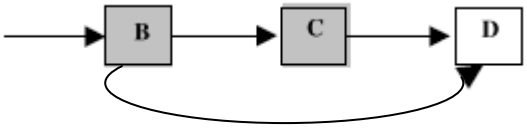


Şekil 3.5: Rota arabelleğinin kullanımını gösteren bir tasarsız ağ örneği.

ii) **Rota talep Mesajlarını Sırtlama (PiggyBacking):** DSR, rota arama gecikmesini düşürmek için rota talep mesajı üzerindeki verileri sırtlayabilir. Eğer hedef durumunda olmayan bir düğüm arabelleğinden bu hedefe giden bir rota bilgisi ile cevapta bulunursa bu düğüm bir veri paketi oluşturma ve bunu hedefe yönlendirerek herhangi bir bilgi kaybının olmamasını sağlama yeteneğine sahip olmalıdır. Bu teknik, talep esaslı protokollerde en büyük sıkıntı olan gecikme problemini azaltma potansiyeline sahiptir.

iii) **Daha Kısa Rotaların Yansıtılması:** İki host birbiriyle arabellekli rotaları kullanarak haberleşiyorken, hostlar birbirine yeteri kadar yaklaştığında bu hostların daha kısa rotaları kullanmaya başlamaları istenir. Birçok durumda temel rota tutma prosedürü bunu başarmak için yeterlidir çünkü eğer hostlardan biri rotanın kısaltılmasına olanak vermeye yetecek kadar hareket ederse muhtemelen mevcut rota üzerindeki ilk sekmenin iletim kapsamı dışına çıkacaktır.

Daha kısa rotaları yansıtma olarak tanımlanan bu yöntem için, eğer hostlar ağ arayüzlerini her paketi alma modunda çalıştırıyorlarsa bir iyileşme söz konusudur. Şekil 3.6'da gösterildiği gibi bir paketin sevk edilmesi içerisinde bir yerlerde mobil host B'nin C'ye bir paket gönderdiğini, D'nin paket rotası içerisinde C'den sonra gelen sekme olduğunu düşünelim. Eğer D bu paketi alırsa paketin, paket rotası ile amaçlandığı gibi kendisine iki yerine bir sekme ile B'den geldiğini görmek için paketi inceleyebilir. Bu durumda D, rotanın, C üzerinden ara sekmeyi devre dışı bırakmak üzere rotanın kısaltılabileceği sonucunu çıkartabilir. D bu durumda paketin ilk göndericisine talep edilmemiş bir rota cevap paketi göndererek şu anda B'den tek sekme ile D'ye ulaşabileceğini bildirir.



Şekil 3.6: Mobil host, rotanın kısaltılabileceğini bildiriyor.

iv) **Ağ Segmentleri:** Tasarsız bir ağ içerisinde ele alınması gereken genel bir hata durumu, ağın etkin bir şekilde segmentlere ayrılmış olması durumudur. Yani haberleşmek isteyen iki host birbirilerinin iletim kapsamı içerisinde değildir ve aralarında, paketleri üzerinden iletebilecekleri bir sekme dizisi oluşturmak üzere yeterli sayıda başka mobil host yoktur. Eğer bu durumda bir host tarafından gönderilen her paket için yeni bir rota arama prosedürü başlatılırsa tasarsız ağın bu host tarafından erişilebilir alt ağı içerisine çok sayıda verimsiz Rota talep paketi yayılır. Bu rota arama prosedürlerinden doğan protokol yükünü azaltmak için aynı hedef için herhangi bir hosttan yeni rota arama prosedürlerinin başlatılabilme oranını sınırlandırmak amacıyla bir üssel geri çekilme algoritması (exponential backoff algorithm) kullanılır.

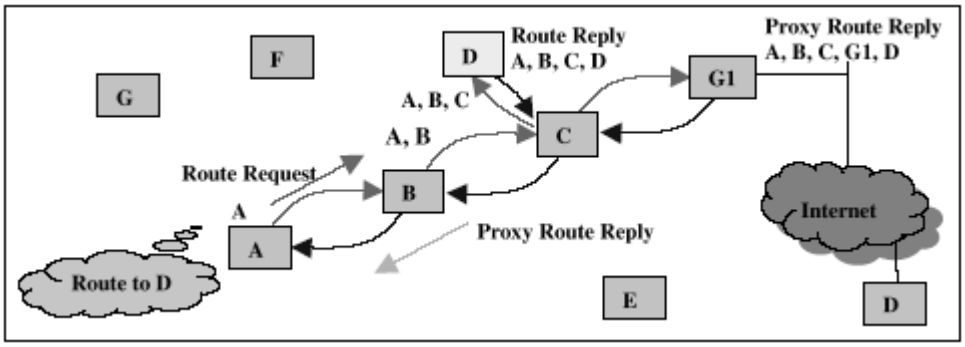
v) **Rota Hata Paketlerinin Gizlice Dinlenmesi (Her Paketi Alma Modu)**: Hataların ele alınması sürecini geliştirmek için mümkün olan bir diğer optimizasyon, hostun, diğer hostlara gönderilen Rota Hata paketlerini gizlice dinlemesine olanak vermek için her paketi alma modu kullanmaktır. Bir Rota Hata paketi, hataya sebep olan rota sekmesinin her iki ucunu tanımladığından hata paketini alan herhangi bir host kendi rota arabelleğini güncelleyerek, paket içerisinde belirtilen iki hostun artık doğrudan haberleşemediğini yansıtabilir. Rota hata paketini alan bir host, bu sekmeyi kullanan rotalar için bir rota arama prosedürü başlatabilir ve bulunan her sekme için rota bu sekmede kırpılır. Hataların ele alınması sürecine yapılan son iyileştirme, bir hostun rota arabelleği içerisinde "negatif" bilgilerin arabelleklenmesinin desteklenmesidir.

vi) **Simetrik Bağlantı Desteği:** DSR her ne kadar asimetrik bağlantıları destekliyorsa da, sadece çift yönlü bağlantıların kullanılmasını sağlayacak şekilde konfigüre edilebilir. Eğer tüm rota talep mesajlarının geldikleri ters yolda üzerinden kaynak düğüme geri dönmesi gerekiyorsa rota talep mesajları tek yönlü bağlantılar üzerinden alınmayacaktır.

vii) **Kurtarma (Salvaging):** Kurtarma, hedef yol üzerindeki ara bir düğümün bir sonraki sekmenin erişilemez durumda olduğunu fark etmesi durumunda meydana gelir. Eğer arabellekte alternatif bir rota varsa kopuk bağlantının yerini alabilir. Fakat bu ara düğümün halen kaynağa bir rota hata mesajı göndermesi gereklidir.

3.2.5. Internet İle Entegrasyon

Maltz vd., Rota Arama ve Rota Tutma mekanizmalarını, tasarsız ağ içerisindeki düğümler ile daha geniş kapsamlı internet kapsamındaki dış düğümler arasında haberleşme olmasını destekleyecek şekilde genişletmiştir. Böylece tasarsız ağ kapsamında yer alan her düğüm, ağ içerisindeki ve dışarısındaki düğümler ile haberleşirken sabit bir kimliğe sahip olur. Burada her düğümün, ev adresi olarak adlandırılan ve tüm diğer düğümler tarafından bilinen tek bir IP adresi seçmesini isteriz. Bu ev adresi, Mobil IP tarafından tanımlanan IP'ye özdeştir. Mobil IP'de olduğu gibi her düğüm kendi ev adresi ile konfigüre edilir ve gönderdiği tüm paketler için bu adresi IP kaynak adresi olarak kullanır.



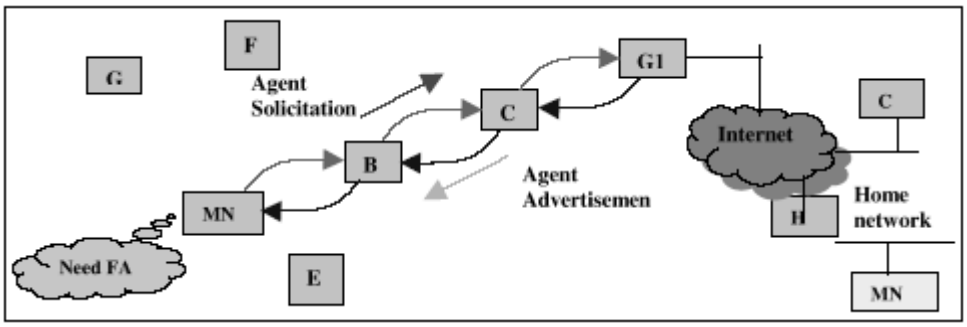
Şekil 3.7: Tasarsız ağ içerisinde bulunmayan bir düğüm için yapılan ve yabancı bir unsur tarafından yanıtlanan rota talebi.

Şekil 3.7, tasarsız ağ içerisinde yer alan ve ağ dışındaki D düğümüne giden rotay arayan A düğümünü göstermektedir. D hedefi için Rota Talebi ağda yayıldıkça en son G1 ağ geçidi düğümü tarafından alınır ve G1, yönlendirme tablosuna başvurur. Eğer D hedefinin tasarsız ağ kapsamı dışında erişilebileceğine kanaat getirirse kendisini rota içerisinde ikinci-son düğüm olarak listeleyen bir Rota cevabı gönderir ve cevabı işaretleyerek A'nın bunu bir proxy cevabı olarak fark etmesini sağlar. Eğer D hedef düğümü fiziksel olarak tasarsız ağ kapsamındaysa A düğümü hem G1'den hem de

D'den bir Rota cevabı alacaktır. A, bunlardan hangisinin proxy cevabı olduğunu ayırt edebilir ve D'ye paket gönderirken doğrudan rotayı tercih edebilir.

3.2.6. Mobil IP İle Entegrasyon

DSR, bir tasarsız ağ ile internet arasında, paketlerin şeffaf olarak tasarsız ağdan internet içerisindeki düğümlere ve internetten tasarsız ağ içerisindeki düğümlere yönlendirilmesine olanak veren kesintisiz karşılıklı çalışma mekanizmasını destekler şekil 3.8'de. MN düğümü, mevcut konumuna bağlı olarak farklı IP alt ağlarına iştirak edebilmek zorunda olduğundan internete bağlanmak için Mobil IP kullanır. MN'nin tasarsız ağa ait olmayan bir alt ağ içerisinde hedefe nasıl yönlendirildiğini, fakat MN'nin tasarsız ağ kapsamında gezindiğini ve G1 düğümünün tasarsız ağ ile internet arasında Mobil IP yabancı sunucu servisleri sağlayan bir ağ geçidi olduğunu şekil 3.8'de görülmektedir.



Şekil 3.8: Tasarsız ağ içerisinde G1 konumunda yer alan yabancı bir sunucuya kaydolan gezgin düğüm.

Mobil düğüm (MN) tipik olarak kendi ağ arayüzünü her paketi alma modunda tutacak ve böylece Rota Talebi, Rota Cevabı veya üzerlerinde DSR kaynak rotalara sahip veri paketleri gibi DSR paketlerine parametre eklerken bir DSR ağına girdiğini bilecektir.
Eğer MN, en iyi bağlantının tasarsız ağ üzerinden olacağına karar verirse IP sınırlı yayın adresi (255.255.255.255)'ni hedefleyen bir Rota talebi üzerinde sırtlanmış bir Mobil IP Sunucu İstemi yayınlar. Bu durum, her ne kadar ağ geçitleri alt ağlar arasında yayılmasına izin vermeyecekse de tasarsız ağ üzerindeki sekmelere İSTEM'in yayılmasını sağlar. G1 noktasındaki yabancı sunucu İSTEMİ aldığında bir SUNUCU İLANI yaparak MN'nin tasarsız ağı ziyaret eden bir Mobil IP düğümü olarak bu yabancı sunucuya ve kendi lokal sunucusuna kendisini kaydetmesini sağlar. Kayıt işlemi tamamlandığında mobil düğümün lokal sunucusu, mobil düğüm MN için

hedeflenmiş paketleri G1 konumundaki yabancı sunucuya tünellemek için bir Mobil IP kullanacak ve G1, bu paketleri lokal olarak DSR'yi kullanan mobil düğüme teslim edecektir.

3.2.7. DSR ile Çoğa Gönderimli Yönlendirme

DSR halihazırda gerçek bir çoğa gönderimli yönlendirme desteklememekte fakat birçok ağ kaynağı içerisinde yeterli olan bir konjektürü desteklemektedir. Rota Arama mekanizmasının bir miktar daha genişletilmesi suretiyle DSR, tasarsız ağ içerisinde, paketi gönderenin tanımlı sekme sayısı kapsamında yer alan tüm düğümlere veri paketinin kontrollü sel algoritması ile gönderilmesi mekanizmasını destekler. Bu düğümler daha sonra, paketi belirtilen çoğa gönderimli hedef adresine kaydolmuş bu düğümlere sınırlandırmak üzere hedef adres filtrelemesi (mesela yazılımsal olarak) uygulayabilir. Her ne kadar bu mekanizma ağ kaynaklarını korumak için yayın ağacının kırpılmasını desteklemiyorsa da, tasarsız ağ içerisinde çoğa gönderimli hedef adresine kaydolmuş tüm düğümlere bilgi dağıtmak için kullanılabilir. Bu mekanizma aynı zamanda gönderici etrafında sınırlı bir alan kapsamındaki tüm düğümlere uygulama düzeyinde paketlerin gönderilmesi için kullanışlı olabilir.

Bu tarz bir çoğa gönderim şeklinde DSR düğümü üzerindeki bir uygulama, çoğa gönderimli hedef adrese bir paket gönderir ve DSR, çoğa gönderimli adrese hedeflenmiş bir Rota Talebi içerisinde paketten verileri sırtlar. Normal Rota Talebi yayma yöntemi, bu paketlerin, paketi gönderenin tanımlı sekme sayısı (TTL) kapsamında ağ içerisindeki tüm düğümlere etkin olarak dağıtılmasını sağlar. Paket, Rota Arama mekanizması için tanımlandığı şekilde sevk edildikten sonra, paketi alan her düğüm münferit olarak hedef adresini inceler ve eğer bu paket kendisinin kayıtlı olmadığı bir çoğa gönderimli adrese yönlendirilmişse paketi dikkate almaz.

3.3. HEDEF SIRALI UZAKLIK VEKTÖRÜ (DSDV)

Hedef Sıralı Uzaklık Vektörü Yönlendirmesi (DSDV), tasarsız ağlara ilave edilen geleneksel bir yönlendirme protokolüdür. Klasik bir uzaklık vektörü algoritması olan Dağıtık Bellman-Ford (DBF) algoritmasından doğmuştur. Temel DBF içerisindeki döngü problemini gidermeye yönelik iyileştirmeler yapılmaktadır. Döngü oluşumu, yönlendirme bilgisini çağırmak üzere sıra numarasına sahip her rota tablosu kaydını etiketlemek suretiyle giderilir.

3.3.1. Protokole Genel Bakış

DSDV'de paketler, her düğümde saklanan yönlendirme tabloları kullanılarak bir tasarsız ağın düğümleri arasında yönlendirilir. Her yönlendirme tablosu, her düğümde, mevcut tüm hedefleri ve bu hedeflerden her birine kadar olan sekme sayısını listeler. Her rota tablosu kaydı, hedef tarafından oluşturulmuş bir sıra numarası ile etiketlenir. Yönlendirme tablolarının tutarlılığını sağlamak için DSDV hem periyodik hem de tetiklemeli yönlendirme güncellemeleri kullanır. Ağ içerisinde herhangi topolojik bir değişim olduğunda yönlendirme bilgisini mümkün olduğunca çabuk yaymak için periyodik güncellemelere ilave olarak tetiklemeli yönlendirme güncellemeleri kullanılır. Güncelleme paketleri içerisinde her düğümden erişilebilir olan hedefler ve her hedefe ulaşmak için gerekli olan sekme sayısı ile her rota ile ilişkilendirilmiş olan sıra numarası yer alır. Her hedef için ayrıca ilk rota varışı ile en iyi rota varışı arasındaki zaman farkı hakkında veriler de tutulur. Bu verilere dayalı olarak, değişmeye yüz tutan ve bu sebeple rota tablolarında oynamalara neden olacak rotaların ilan edilmesini geciktirmeye yönelik bir karar alınabilir. Normalde aynı sıra numarası ile ulaşan muhtemel rota kayıtlarının yeniden yayınlanma sayısını düşürmek için muhtemelen stabil olmayan rotaların ilanı geciktirilir.

3.3.2. Rotanın İlan Edilmesi

DSDV protokolü, her mobil düğümün, kendi rota tablosunu mevcut komşularından her birine ilan etmesini ister. Bu ilan, liste içerisindeki kayıtlar zaman içerisinde dinamik olarak değiştikçe her bir mobil düğümün hemen hemen her zaman bu grup içerisindeki diğer her bir mobil düğümün yerini tespit edebilmesini sağlayacak yeterlilikte yapılmalıdır. Ayrıca her mobil düğüm, talep üzerine diğer düğümlere veri paketi aktarmayı kabul eder. Bu mutabakat, bir hedefe giden rota için en az sekme sayısını tespit etme yeteneğine ek yük bindirecektir. Bu sayede mobil bir düğüm, grup içerisinde bir başka mobil düğüm ile veri alışverişinde bulunabilir, hatta verinin hedefi doğrudan haberleşme kapsamı içerisinde yer almasa da bu alışveriş gerçekleşebilir. Kendi aralarında veri yolları oluşturmak üzere ortak karşılıklı çalışma içerisinde bulunan tüm düğümler, her beş saniyede bir periyodik olarak gerekli verileri yayınlar.

3.3.3. Rota Tablosu Kaydı ve Rota Seçim Kriterleri

Her düğüm tarafından yapılan veri yayını, her yeni rota için yeni sıra numarasını ve şu bilgileri ihtiva edecektir:

- Hedefin adresi
- Hedefe ulaşmak için gerekli olan sekme sayısı
- Hedef tarafından ilk durumda damgalandığı üzere hedefe ilişkin olarak alınan bilginin sıra numarası

Bir X düğümü ortaya çıktığında, ilk olarak bir işaret mesajı ("yaşıyorum mesajı") yayınlar. Bu mesaj düğüme lokal olarak tutulan sıra numarasını damgalar. Bu düğüme komşu olan düğümler bu mesajı dinler ve bu düğüm için bilgileri günceller. Eğer düğümler bu X düğümü için herhangi bir eski kayda sahip değilse yönlendirme tabloları içerisine hemen X'in adresini ve ayrıca sekme sayısın (bu durumda 1) ve sıra numarasını, yayınlanan X olarak işler. Eğer düğümler X için daha eski bir kayda sahipse yayınlanan bilgilerin sıra numarası, X hedefi için düğüm içerisinde saklanan sıra numarası ile karşılaştırılır. Eğer alınan mesaj daha yüksek bir sıra numarasına sahipse o halde düğüm, konumu hakkında yeni bilgileri ağ geneline yaymış demektir ve bu yüzden alınan yeni bilgi gereğince kaydın güncellenmesi gereklidir. Daha yeni bir sıra numarasına sahip bilgi, X düğümü bizzat sıra numarasını damgaladığından kesinlikle yeni bir bilgidir.

Bir düğümün aldığı yeni bilgi, bu düğümün komşularına yayınlanmak üzere programlanır, böylece komşular topolojideki değişimleri öğrenmiş olur. Komşu düğümler de yine aynı kuralı takip eder, yani yeni bir sıra numarasına sahip bir düğüm hakkında bilgi alındığında bilgilerini günceller. Yeni alınan yayın bilgisinden seçilen rotalar için metrik değerler, her sekme için bir birim arttırılır, böylece yeni bilgiler tüm düğümlerde kademeli olarak güncellenir ve bu düğümler, paketi X hedefine doğru şekilde yönlendirmek için bir sonraki sekmeyi bilir hale gelir.

3.3.4. Topoloji Değişimlerine Cevap Verme

Mobil düğümler, bir yerden başka yere hareket ettikçe kopuk bağlantılar yaratabilir. Bir sonraki sekmeye giden bir bağlantı koptuğunda, bu sekme üzerinden giden herhangi bir rotaya hemen sonsuz bir metrik değer ve güncellenmiş bir sıra numarası atanır. Bu durum sadece, hedef dışında sair bir kişi tarafından sıra numaralarının tahsis edilmesi durumunda gerçekleşir. Bir düğüme sonsuz bir metrik değer ulaştığında ve bu düğüm, kendisine ulaşan sıra numarasına eşit veya bundan daha yüksek bir sıra numarasına sahip olduğunda bir rota güncelleme yayını tetiklenir. Böylece sonlu metrik değere

sahip rotaların yerini hemen, yeni tespit edilen hedeften yayılan gerçek rotalar alacaktır. Topoloji değişimleri olduğunda düğümlerin ve komşularının yanıltıcı sıra numaraları üretmesini önlemek için düğümler sadece kendileri için sıra numaraları üretir ve bağlantı değişimlerine cevap veren komşular sadece tek sıra numaraları üretir.

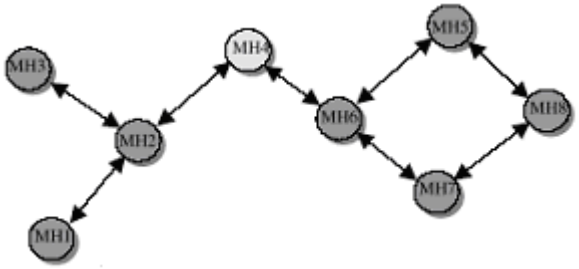
Ağ üzerindeki protokol yüklerini azaltmak için bir düğüm iki farklı paket tipi seçebilir: Tam döküm (full dump) ve marjinal (incremental). Tam döküm mesajları, göndericinin yönlendirme tablosundan mevcut tüm yönlendirme bilgilerini taşır. Bu bilgiler muhtemelen birden fazla pakete ihtiyaç duyar. Marjinal mesajlar ise sadece yapılan son tam içerik dökümünden bu yana değişen bilgileri taşır. Bu mesajlar tek bir paket ile sınırlıdır. Düğümler çok sık hareket etmediğinde marjinal dökümler muhtemelen yeterli olacak ve tam dökümler daha nadir olarak iletilecektir. Düğümler sık sık hareket ettiğinde marjinal güncelleme mesajının boyutu tam döküm mesajının boyutuna yaklaşır. Bu durumda tüm dökümler, marjinal güncellemelerin büyüklüğünü azaltmak amacıyla daha sık yapılmak üzere programlanmalıdır.

3.3.5. Baz İstasyonu Kapsamının Genişletilmesi

Mobil düğümler, kendilerine kablolu ağa bağlı diğer düğümler ile veri alışverişi yapma olanağı veren baz istasyonları ile sık sık etkileşimde bulunacaktır. DSDV protokolüne iştirak etmek suretiyle baz istasyonları, kablosuz vericileri tarafından belirtilmiş olan kapsama alanlarını mobil bir düğüm kadar genişletebilir. Bu sayede bir baz istasyonunun kapsamı içerisinde bulunan mobil düğümler, baz istasyonunun kapsama alanını bu alan dışında yer alan diğer düğümlere, bu diğer mobil düğümler kapsama alanı içerisinde yer alan bir mobil düğüme yakın olduğu müddetçe, hizmet vermek amacıyla etkin olarak genişletmek üzere işbirliği yapabilir.

3.3.6. Çalışan DSDV Örnekleri

Tablo 3.1, Şekil 3.10'da gösterilen ağ topolojisi için MH4'te tutulan yönlendirme tablosunun olası bir yapısını göstermektedir. Her düğümün adresini MH_i kabul edersek tüm sıra numaraları $SNNN\_MH_i$ olarak gösterilmektedir. Kurulum zamanı, eski bir rotanın ne zaman silinmesi gerektiğinin tespitine yardımcı olmaktadır.



Şekil 3.9: Bir Tasarsız Ağ örneği.

Tablo 3.1: MH4 Yönlendirme tablosu.

Destination	Next Hop	Metric	Seq. Number	Install	Stable
MH_1	MH_2	2	$S406\_MH_1$	$T001\_MH_4$	$Ptr1\_MH_1$
MH_2	MH_2	1	$S128\_MH_2$	$T001\_MH_4$	$Ptr1\_MH_1$
MH_3	MH_2	2	$S564\_MH_3$	$T001\_MH_4$	$Ptr1\_MH_1$
MH_4	MH_4	0	$S710\_MH_4$	$T001\_MH_4$	$Ptr1\_MH_1$
MH_5	MH_6	2	$S392\_MH_5$	$T002\_MH_4$	$Ptr1\_MH_1$
MH_6	MH_6	1	$S076\_MH_6$	$T001\_MH_4$	$Ptr1\_MH_1$
MH_7	MH_6	2	$S128\_MH_7$	$T002\_MH_4$	$Ptr1\_MH_1$
MH_8	MH_6	3	$S050\_MH_8$	$T002\_MH_4$	$Ptr1\_MH_1$

Courtesy from Carleton university lecture note (94.581x)

Tablo 3.1'den, tüm düğümlerin, hemen hemen aynı zamanda MH4 için elverişli olduğunu, çünkü birçoğunun Kurulum zamanının yaklaşık aynı olduğunu görebiliriz. Tablo 3.2, MH4'ün ilan edilen rota tablosunun yapısını göstermektedir

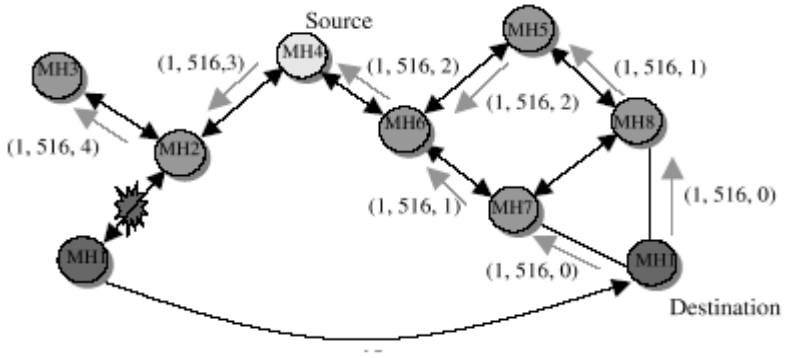
Tablo 3.2: MH4 İlan edilen rota tablosu.

Destination	Metric	Seq. Number
MH_1	2	$S406\_MH_1$
MH_2	1	$S128\_MH_2$
MH_3	2	$S564\_MH_3$
MH_4	0	$S710\_MH_4$
MH_5	2	$S392\_MH_5$
MH_6	1	$S076\_MH_6$
MH_7	2	$S128\_MH_7$
MH_8	3	$S050\_MH_8$

Tablo 3.3: İlan edilen rota (güncellenmiş).

Destination	Metric	Seq. Number
MH_4	0	$S820\_MH_1$
MH_1	3	$S516\_MH_2$
MH_2	1	$S238\_MH_3$
MH_3	2	$S674\_MH_4$
MH_5	2	$S502\_MH_5$
MH_6	1	$S186\_MH_6$
MH_7	2	$S238\_MH_7$
MH_8	3	$S160\_MH_8$

Şimdi MH_1'in MH_5 ve MH_7 civarına hareket ettiğini ve diğerlerinden uzaklaştığını düşünelim (özellikle MH_2'den)



Şekil 3.10: Tasarsız bir Ağ içerisindeki hareketler.

Yukarıdaki Şekil, MH1 hedefi için rota bilgisinin yayılmasını göstermektedir. Her satır hedef adresi, sıra numarasını ve bağlantı metrik değerini göstermektedir.

Rotanın ilanı hedef düğümde başlar ve tüm ağ geneline yayılır. Şekil 3.4, ağ üzerinde bir hedef düğümden diğer düğümlere rota kaydının mantıksal yayılımını göstermektedir. Hedef düğüm, yeni rota ilanlarına bir sıra numarası ekler. Şekil, sadece MH_1 hedefine ilişkin bilgileri göstermektedir. Gerçekte her düğüm, sadece MH_1 bilgilerini değil kendisine ait tüm rota tablosunu yayınlayacaktır. Daha sonra Tablo 3.4'te gösterildiği gibi MH_4'te yeni dahili yönlendirme tabloları belirebilir.

Tablo 3.4, sadece MH_1 kaydını ve yeni bir metrik değer göstermektedir, fakat müdahale zamanında birçok yeni sıra numarası kayıtları ulaşmıştır. Bu değişim sadece yüksek sıra numarasından ötürü meydan gelir. Sırasıyla MH_5 ve MH_7'den aynı güncelleme mesajı (yani aynı hedef ve sıra numarası) MH_6'ya ulaşır. MH_6, metrik değeri daha iyi olduğu için MH_7'den gelen mesajı seçer.

Tablo 3.4: MH4 Yönlendirme Tablosu (güncellenmiş).

Destination	Next Hop	Metric	Seq. Number	Install	Stable Data
MH_1	MH_1	3	S516_MH_1	T810_MH_4	Ptr1_MH_1
MH_2	MH_2	1	S238_MH_2	T001_MH_4	Ptr1_MH_1
MH_3	MH_2	2	S674_MH_3	T001_MH_4	Ptr1_MH_1
MH_4	MH_4	0	S820_MH_4	T001_MH_4	Ptr1_MH_1
MH_5	MH_6	2	S502_MH_5	T002_MH_4	Ptr1_MH_1
MH_6	MH_6	1	S186_MH_6	T001_MH_4	Ptr1_MH_1
MH_7	MH_6	2	S238_MH_7	T002_MH_4	Ptr1_MH_1
MH_8	MH_6	3	S160_MH_8	T002_MH_4	Ptr1_MH_1

MH_4'te, ilan edilmiş marjinal yönlendirme güncellemesi Tablo 4'te gösterilen biçime sahiptir. Bu ilan içerisinde ilk olarak MH_4 bilgisi gelir çünkü ilanı yapan sekme bu sekmedir. Bunu MH_1 bilgisi takip eder çünkü etkide bulunan önemli rota değişimlerine sahip tek sekmedir

3.3.7. Dalgalanmaların Söndürülmesi

DSDV, rota tablosu güncellemeleri içerisindeki dalgalanmaları söndürmek için özel bir mekanizma kullanır. Birçok bağımsız düğümün asenkron olarak yönlendirme bilgisi ilettiği bir ortamda bazı dalgalanmalar meydana gelebilir. Mesela bir düğüme, aynı sıra numarasına sahip ve aynı hedefe giden iki rota ulaşabilir, bu durumda metrik değeri en kötü olan rota her zaman ilk ulaşır. Bu durum, rota güncellemelerinde sürekli kesintilere ve yönlendirme tablolarında dalgalanmalara yol açabilir. DSDV bu problemi, "stabilizasyon süresi" fikrini kullanarak çözmektedir. Rota stabil hale gelene kadar geçen zaman dilimi (stabilizasyon süresi) tahmin edilir ve ağa yeni rota bilgisi ilan edilmeden önce bu stabilizasyon süresinin geçmesi beklenir. Başka bir deyişle stabilizasyon süresi, yeni rotalar ilan edilmeden önce ne kadar bir süre bekleneceğine kadar vermek için kullanılır. Stabil olmayan rotaların ilanını geciktirmek suretiyle yönlendirme tablolarındaki dalgalanmalar önlenir ve sonuç olarak rota güncelleme sayısı azaltılır.

3.3.8. DSDV Protokolü'nün Özellikleri

DSDV teorik olarak şu özelliklere sahiptir:

- Hiçbir an döngü oluşumu olmadan yönlendirme
- Dinamik, çok sekmeli, kendi kendine başlayan nitelikte
- Düşük bellek ihtiyacı
- Tetiklenmiş güncellemeler vasıtasıyla hızlı yakınsama

- Tüm hedefler için elverişli rotalar
- Hızlı işlem süresi
- Makul ağ yükü
- Minimum rota değişimi
- “Mobilite faktörüne” dayalı oalrak 100’e kadar mobil düğümün çalışmasına destek

3.3.9. Döngüye Meydan Vermeyen Mimarinin İspatı

i düğümü sonraki sekmesini her değiştirdiğinde potansiyel olarak bir döngü meydana gelebilir. Bu iki durumda oluşabilir:

* i düğümü bir sonraki sekmesine giden bağlantının kopuk olduğunu algılar: Açık bir biçimde bu işlem, i düğümünü ihtiva eden bir dönü meydana getiremez.

* i düğümü komşularından biri olan k düğümünden D hedefine giden bir rota alır ve bu rotanın sıra numarası s(k) i düğümünde ilk durumda kayıtlı olan sıra numarası s(i) ile aynı olur.

Bir i düğümü s(i) sıra numarasını komşularına bunu ancak bir sonraki mevcut sekmesinden aldıktan sonra yayar. i düğümü, belirli bir hedef için bir sonraki sekme olarak belli bir komşusunu seçtiğinde bunun bilgisi sadece bir sonraki sekme tarafından alınır. Böylece her zaman bir sonraki sekmede kayıtlı olan sıra numarası i düğümünde kayıtlı değere eşit veya bu değerden yüksek olur. i düğümünden başlayarak eğer sonraki sekme imleçleri zincirini takip edersek ziyaret edilen düğümlerde kayıtlı olan sıra numaraları bir azalmayan sıra oluşturacaktır. Şimdi i düğümünün, sonraki sekmesi olarak k’yi seçmesi ile bir döngü meydana getirdiğini düşünelim. Bu durum, i düğümünün k’nın hem öncesinde hem de sonrasında yer aldığı anlamına gelecektir. Bahsedildiği gibi k’nın sonrasında bulunduğundan s(k) <= s(i) olmalıdır (bir yol boyunca sıra numaralarının azalmayan bir sıra meydana getirdiği ispatlandığı üzere). Fakat bu da ilk durumdaki s(k) > s(i) varsayımı ile çelişmektedir. Bu yüzden eğer düğümler, rotaları seçmek için yeni sıra numaraları kullanıyorsa döngü oluşumu meydana gelemez.

3.4. UZAKTAN VEKTÖR YÖNLENDİRME GÜVENLİĞİ

3.4.1. SEAD Temel Tasarımı

Güvenli yönlendirme protokolü SEAD tasarımımızı güvensiz DSDV geçici ağ yönlendirme protokolü DSDV [6] üzerine kuruyoruz. Özellikle, SEAD'daki uzun ömürlü yönlendirme tekrarlarını engellemek için, DSDV'de olduğu gibi hedef sekans numaraları kullanıyoruz; bu hedef sekans numaralarını aynı zamanda SEAD'daki yönlendirme güncelleme mesajlarını tekrardan korumak içindir.

DSDV'den farkımız, tetiklenmiş güncelleştirmeler gönderken ortalama ağırlıkta bir çökme zamanımızın olmamamızdır.

Lüzumsuz tetiklenen güncelleştirmelerin sayısını azaltmak için, DSDV kulvarlarındaki her düğüm, her hedef için, düğümün bu hedef için yeni bazı dizi numaraları için ilk güncelleştirmeyi alması ile onun için dizi numarası için en iyi güncelleştirmeyi alması arasında geçen ortalama zaman (alınanlar arasında minimum ölçü ile); tetiklenmiş bir güncelleme yollamaya karar verirken, her DSDV düğümü, ortalama ağırlıklı yerleşme zamanı için bir hedef için her bir tetiklenmiş güncellemeyi, bu dizi numarası için en iyi ölçü ile tetiklenmiş bir güncellemeye ihtiyaçç duyma umudu ile, geciktirir.

SEAD kötü niyetle gecikme kullanmayabilen düğümlerden gelen saldırıları önleyebilmek için, böyle bir gecikme kullanmaz. Bir düğüm aldığı ilk rotayı en yüksek dizi numarası ve en düşük ölçü ile seçtiğinden, bir saldırgan aksi takdirde, kendi tetiklenmiş güncellemelerinde gecikmeyi engelleyerek, kendisi üzerinden gönderilen daha fazla trafiğe sebep olmaya çalışabilir. Bu tip bir saldırı başka türkü saldırganı gizlice dinleme pozisyonuna, modifiye etme, veya başka bağlı paketlerini modifiye etmeye meyilli kılar.

Ek olarak, DSDV'nin aksine, bir düğüm bir hedefe durak bağlantısının bozulduğunu anladığında ve bu girişte ölçüyü sonsuz olarak girdiğinde, düğüm bu hedef için dağıtım tablosunda dizi sayısını arttırmaz. Yüksek dizi sayıları öncelikli olduğundan, bu düğümün yönlendirme güncellemesi bu yeni dizi sayısı ile doğrulanmış olmalıdır. Ancak bu daha büyük dizi sayılarını doğrulamak için bir mekanizma içermedik. Bunun yerine, bu aynı dizi sayısı için yeni bir güncelleme kabul etmemek için düğüm bu hedef için yönlendirme tablosunu işaretler. Bu aksi takdirde bu durumda oluşabilecek geleneksel uzaklık vektörü "sonsuza sayım" sorununu [14,28] ve mümkün olan yönlendirme tekrarını etkin bir şekilde önler.

3.4.2. Tek Yönlü Sağlama Zincirleri

Bir tek yönlü sağlama zinciri, tek yönlü sağlama fonksiyonu üzerine inşa edilmiştir. Standart bir sağlama işlevi gibi, tek yönlü sağlama fonksiyonu, H, herhangi bir uzunluk girişini sabit uzunlukta bir bit dizgisine eşlemler. Dolayısıyla, H: {0, 1} * $\{0,1\}^{\rho}$ ρ sağlama fonksiyonunun çıkışının bit uzunluğudur. H fonksiyonu hesaplaması kolay olmalı, diğer taraftan sayısal olarak dönüştürülmesi genel olarak olanaksız olmalıdır. Tek yönlü sağlama fonksiyonlarının daha biçimsel bir tanımı Goldwasser ve Bellare (11) tarafından sağlanmış ve bu tip bir dizi fonksiyon önerilmiştir. MD5 (44) ve SHA-1 (32) de buna dahildir.

Tek yönlü sağlama zinciri oluşturmak için, bir düğüm rastgele bir başlangıç değeri seçer; x 2 f0; 1gq ve tek yönlü bir sağlama zinciri oluşturmak için değerler listesini hesaplar, bir düğüm rastgele bir başlangıç değeri seçer x 2 $\{0; 1\}^{\rho}$ ve değerler listesini hesaplar h0, h1, h2, h3,. . . , h*n* burada h0 = x, ve 0 < I ≤ n için hi = H(hi-1), bazı n için. Başlatma sırasındaki düğüm, yukarıda gösterildiği gibi sağlama zincirinin elemanlarını oluşturur. "Soldan sağa" (artan altsimge i'nin sırasıyla) ve daha sonra yönlendirme güncellemelerini sabitlemek için zincirin bazı elemanlarını kullanır; bu değerleri kullanırken, düğüm oluşturulmuş zincirde "sağdan sola" ilerler (azalan altsimge i'nin sırasıyla).

Tek yönlü sağlama zincirinin mevcut doğrulanmış unsuru göz önüne alındığında, elemanları daha sonra zincir içerisinde kullanılan dizide doğrulamak mümkündür (daha çok "sola" doğru, veya azalan altsimgenin sırasıyla). Örneğin, doğrulanmış bir hi değeri verildiğinde, bir bağ H(H(H(hi-3))) hesaplayarak hi-3'ü doğrulayabilir ve çıkan değerin h*i'ye eşit olduğunu* gerçekleyebilir.

Doğrulama amacıyla tek yönlü sağlama zincirleri kullanmak için, bağın oluşturulmuş sağlama zincirinden h*n* gibi bir otantik eleman dağıtması için bir mekanizma varsayalım. Bu anahtar dağıtım için geleneksel bir yaklaşım, güvenilir bir varlığın ortak-anahtarını kullanarak kendisi için yeni bir sağlama zincir elemanı işaret etmesidir. Güvenilir bir ortak anahtar altyapısı üzerine [19] dayanmadan, Hubaux, Buttyan ve Capkun PGP benzeri sertifikalardan güven ilişkileri önyükler. Alternatif olarak, güvenilir bir düğüm, güvenli şekilde sadece simetrik-anahtar şifreleme kullanarak

[17,39] veya şifreleme olmayan yaklaşımlarla doğrulanmış bir sağlama zinciri elemanı dağıtabilir [46].

Bir düğüm SEAD'de tek yönlü bir sağlama zincirinden m grupları şeklinde elemanlar kullandığından, bir düğümün, n, m'e bölünebilir olacak şekilde kendi sağlama zincirini oluşturduğunu farz ederiz. Bir düğüm ağa ilk kez girdiğinde, veya bir düğüm sahip olduğu sağlama zinciri elemanlarının çoğunu kullandıysa, rastgele yeni bir x alabilir, bu x'den yeni bir sağlama zinciri oluşturabilir ve yeni üretilmiş h*n* değerini yukarıda tanımlandığı gibi güvenilir bir varlığa veya alternatif bir doğrulama ve dağıtım servisine gönderir.

3.4.3. Ağaç Doğrulanma Değerleri

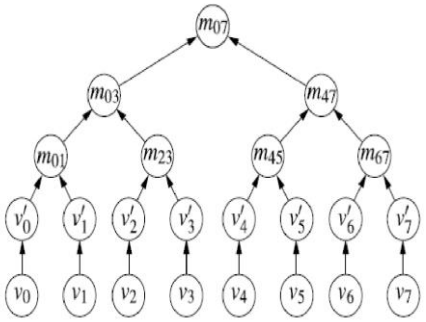
Ağaç doğrulanmış değerler mekanizması ilk kez Merkle tarafından sunulmuş ve Merkle sağlama ağaçları olarak da bilinen etkili bir sağlama ağacı doğrulama mekanizmasıdır [30]. v*0*; v*1*; . . . ; v*w-1* değerlerini doğrulamak için, bu değerleri bir ikili ağacın yaprak düğümlerine yerleştiririz. (Kolaylık olsun diye, w ikinin bir kuvveti olacak şekilde dengeli bir ikili ağaç varsayıyoruz). v' *i* = H[vi] olacak şekilde, tüm vi değerlerini, açıklayıcı komşu değerleri engellemek için doğrulanan bilgide bir tek-yönlü sağlama fonksiyonu H ile kapatıyoruz. Daha sonra v`*0*; . . . ; v` w-1 değerlerine teslim etmek için Merkle sağlama ağacı yapısını kullanıyoruz. İkili ağacın her bir iç düğümü kendi iki alt düğümlerinden çıkarsanmıştır. ml ve mr: mp ¼ H½mlkmr_ (k birleştirme belirtir) sağ ve sol alt düğümlerden ana düğüm türetmeyi hesaba katmak gerekir.

Ağacın değerlerini yinelemeli olarak yaprak düğümlerinden kök düğüme hesaplarız. Şekil 1, v0; v1; . . . ; v7, e.g., *m01* =H(v'0 IIv' 1), m03 = H[m01I m23] sekiz değer üzerinden bu yapıyı gösterir.

Ağacın kök değeri ağacın tümünü işlemek için kullanılır, ve ek bilgi ile birlikte, herhangi bir yaprak değeri doğrulamak için kullanılabilir. Bir vi değerini doğrulamak için gönderen, i, vi ve vi'den kök düğümüne tüm düğümlerin tüm kardeş düğümlerini, gelen kök düğüme yolunda düğümleri açığa çıkartır. Alıcı daha sonra bu düğümleri, vi değerini doğrulayan köke giden yolu doğrulamak için kullanabilir, Alıcı bu düğümleri köke giden yolu doğrulamak için kullanabilir. Örneğin, bir alıcı Şekil 1'deki v2 anahtarını doğrulamak istiyorsa, pakette v0 3;m01;m47 değerlerini de içerir. Otantik bir m07 kök değeri ile bir alıcı H[H[m_{01}IIH[H[$_{V2}$]IIv`3]]IIm$_{17}$] kaydedilmiş m07 kök

değerine eşit olduğunu doğrulayabilir. Doğrulama başarılı ise, alıcı v2'nin otantik olduğunu bilir.

Şekil 1'deki v' 0; v' 1; . . . ; v' 7, ağaca (bu örnekte) v2'yi doğrulamak v3 değerinin açığa çıkmasına engel olmak için eklenmiştir.



Şekil 3.11: Ağaçla doğrulanmış değerler.

3.4.4. Ölçü ve Dizi Numarası Doğrulayıcıları

Bizim SEAD protokolümüz arasındaki farklara ek olarak, SEAD içindeki bir yönlendirme güncellemesindeki her ölçü üzerindeki alt sınır doğrulama ile güvence altına alınır. Buna ek olarak, SEAD yönlendirme bilgisinin alıcısı aynı zamanda göndericiyi de doğrular (yönlendirme bilgisinin doğru göndericiden kaynaklandığını garanti altına alır). Uzaklık ölçüsündeki alt sınırın doğrulanmasını bu bölümde tanımlıyoruz ve komşu doğrulamasını bir sonraki bölümde tanımlıyoruz. Böylelikle SEAD bu saldırılara direnmektedir. Her hangi bir diğer düğümde doğru olmayan yönlendirme durumu yaratan çoğul koordine olmamış saldırganlara karşı, hatta ağdaki aktif saldırganlara ve tehlikeli düğümlere rağmen sağlamdır. Uzak vektör yönlendirmesi protokolündeki yönlendirme güncellemelerini doğrulamak için kullanılabilecek olası bir yaklaşım her düğümün yönlendirme güncellemelerini asimetrik şifreleme kullanarak işaretlemesidir. Bununla birlikte bu yaklaşım bir plansız ağda kullanılmak üzere üç farklı problem ortaya koyar.

İlk olarak, bazı kurban düğümlere yüksek sayıda keyfi taklit edilmiş yönlendirme güncellemesi gönderebilir, öyle ki kurban tüm CPU kaynaklarını bu güncelleme akışlarını doğrulamaya çalışırken harcar bu da etkili bir Hizmeti-Engelleme-Saldırısı yaratır. Bu saldırı bazı plansız ağlarda özellikle kolay olur, nitekim plansız ağ düğümleri telli ağdaki iş istasyonlarından daha güçsüz CPU'lara sahiptir. İkinci olarak,

diğer düğümlerin saldırgana yönelik bu hedef düğümü için yanlışlıkla doğrudan paketler göndermesine yol açacak şekilde, bir düğümle uzlaşan bir saldırgan diğer her düğümün komşu (ölçü 1) olduğunu öne sürerek güncellemeler gönderebilir.

Son olarak, hiç saldırgan mevcut olmasa bile, asimetrik şifrelemenin daha uzun imzaları ve daha uzun imza oluşturma ve doğrulama zamanı, başka türlü kullanışlı uygulamalar yürütmek ve kullanışlı iletişim kurmak için kullanılabilecek kaynakları azaltır. Bu problem geçici bir ağda geleneksel bir ağa göre (yani telli ve sabit) geçici ağdaki sınırlı düğüm ve bağlantı kaynağı nedeniyle (mevcut bant genişliği, CPU kapasitesi, ve pil gücü/enerjisi gibi) daha ciddidir.

Bunun yerine, SEAD içinde yönlendirmeyi güvence altına almak için etkili tek-yönlü sağlama zincirleri kullanıyoruz [26]. Tek-yönlü bir sağlama zincirinin temel operasyonu tanımlanmıştır. SEAD'daki her düğüm, her yönlendirme güncellemesinde kendi sağlama zincirinden kendi hakkında gönderi yapan (metric 0) belirli bir bitişik tekil eleman kullanır. Bu ilk eleman esas alınarak, tek-yönlü sağlama zinciri kavramsal olarak bu hedef için başka yönlendirme güncellemelerinde ölçünün alt sınırı için doğrulama sağlar. Doğrulama ölçüde sadece bir alt sınır sağlar, nitekim kötü niyetli bir düğümü bu rotayı duyduğu düğümle aynı ölçüyü talep etmekten alı koyamaz. Özellikle, tek-yönlü sağlama fonksiyonu, başka bir düğümün yönlendirme güncellemesindeki bir ölçüyü sadece yükseltebilmesi ama alçaltamaması özelliğini sağlar. Tek-yönlü sağlama fonksiyonunun özelliklerine göre, sağlama zincirine hangi değer verilirse verilsin, saldırgan kendi hakkında gönderi yapan gelecekteki bir güncellemede bu düğüm tarafından kullanılan zincirde bir değer yaratamaz (zincirde verili değerin "soluna" doğru daha küçük bir altsimgeli değer). Benzer şekilde, kendi yönlendirme güncellemesindeki başka bir hedefe doğru bir rota tanımlayan her giriş için, bu hedefin sağlama zinciri bu girişteki ölçünün onu kabul eden düğümler tarafından doğrulanmasına izin verir. Belirtildiği gibi, bir alt sınırın bir geçici ağın çapına yerleştirilebileceğini varsayıyoruz, ve bu bağı m -1 ile simgeliyoruz. Böylelikle yönlendirme protokolü içinde, herhangi bir yönlendirme güncellemesindeki tüm ölçüler m'den azdır. SEAD tarafından bir yönlendirme güncellemesindeki bir girişi doğrulamakta kullanılan yöntem, bu hedef düğümün sağlama zincirinden bir m elemanları bitişik grubunu belirlemek için bu girişteki dizi sayısını kullanır. Bu elemanlar grubu içinden girişi doğrulamak için kullanılması gereken belirli eleman bu

girişte gönderilen ölçü değeri tarafından belirlenir. Özellikle, eğer bir düğümün sağlama zinciri şu değerler dizisi ise
h*0*; h*1*; h*2*; h*3*; . . . ; h*n*

ve n m'e bölünebilir ise, bazı yönlendirme güncelleme girişindeki bir dizi sayısı için şu geçerlidir: k (n/m)- i.
Bu sağlama zincirindeki h_{km}; h_{km1}; . . . ; h_{km+m_1} elemanlar grubundan bir eleman girişi doğrulamak için kullanılır, eğer bu giriş için ölçü değeri j; 06j < m ise, buradaki hkmj değeri bu dizi sayısı için yönlendirme güncellemesi girişini doğrulamakta kullanılır. SEAD'deki bir düğüm bir yönlendirme güncellemesi gönderirse, düğüm bu güncellemedeki her bir girişle birlikte bir sağlama değeri içerir. Bu güncellemede eğer düğüm kendisi için bir giriş listelerse, bu girişteki adresi kendi düğüm adresine, ölçüyü 0'a, dizi sayısını sıradaki kendi dizi sayısına, sağlama değerini bu dizi sayısına denk gelen kendi sağlama zinciri elemanları grubundaki ilk elemana ayarlar. Dizi numarası i için yukarıda verilen örnekte, düğüm bu girişteki sağlama değerini kendi h_{km}'sine ayarlar. Bu güncellemede düğüm bir başka hedef için bir giriş listelerse, bu girişteki adresi hedef düğümünün adresine ayarlar, ölçü ve dizi sayısını kendi yönlendirme tablosundaki hedef için değerlere ve sağlama değerini bu hedefe yönelik rotayı öğrendiği yönlendirme güncellemesi girişinde kabul ettiği sağlama değerine ayarlar.
Bir yönlendirme güncellemesi girişindeki dizi sayısına ve ölçüye uygun olarak herhangi bir düğümün bir rotayı, sağlama zincirinin tek-yönlü doğasına göre hedefin kendi güncel dizi sayısından daha büyük bir dizi sayısı talep eden herhangi bir hedefe duyurmasını engeller. Aynı şekilde, herhangi bir düğüm bir rotayı ilan aldığı diğerlerinden daha iyi ilan edemez, çünkü var olan bir rotadaki ölçü azaltılamaz. Tanımlandığı gibi, aynı sağlama zincirinden daha önceki herhangi bir otantik sağlama elemanı verilmesi koşuluyla, Yönlendirme güncellemesi alan düğümler güncellemedeki her girişi kolaylıkla doğrulayabilir. Yüksek bir dizi sayısı talep eden kötü niyetli bir güncellemenin, güncellemeyi doğrulamak için bir alıcı düğümünü yüksek sayıda sağlama operasyonu gerçekleştirmeye zorladığı saldırılara karşı korumak için, bir alıcı düğüm benzeri her bir doğrulama için gerçekleştirmek istediği sağlamaların sayısını sınırlayabilir, çünkü DSDV (ve bu nedenle SEAD) ağda yeni yönlendirme bilgisi yayar ve bu sınır, otantik bir güncelleme alınmadan önce alıcı düğümün kaçırdığı bir hedef hakkındaki yönlendirme güncellemelerine bir bağımlılık varsayar. Bu tip bir saldırıya

benzer bir çözüm, her düğümün kendi dizi sayı kuşağını gevşekçe senkronize edilmiş bir saat değerine bağlaması ve bu şekilde bir alıcı düğüme, bir güncellemedeki talep edilen dizi sayısının olayı doğrulamak için kastedilen sağlamaları gerçekleştirmeden önce otantik olup olmayacağını belirlemesine izin verir.

Bir düğüm bir yönlendirme güncellemesi aldığı zaman, bu güncelleştirmedeki her giriş için, bu düğüm tarafından bu hedefin sağlama zincirinden aldığı en son öncelikli otantik sağlama değeri ile birlikte, hedef adresini, dizi sayısını, ve alınan girişteki ölçüyü kullanarak, düğüm bu girişteki doğrulamayı kontrol eder. Eğer böyleyse, giriş otantiktir ve düğüm bunu normal alınmış yönlendirme güncelleme girişi olarak yönlendirme algoritmasında işler; aksi halde, düğüm alınan girişi reddeder ve yönlendirme tablosunu buna dayanarak değiştirmez.

Bir saldırgan için dolaşımdaki yönlendirme güncelleme mesajlarını değiştirmek mümkün olabilir, ve bu tip bir saldırgan belirli rotaların ilan edilmesini engelleyebilir; ancak bu tip bir saldırgan tüm yönlendirme güncellemesini de mahvedebilir, bu da bir boğma saldırısına benzer. Protokol ayrıca bir yönlendirme güncellemesinin kaynak adresinin değiştirilmesine karşı ve solucan deliği saldırılarına karşı, MAC düzleminde başka mekanizmalar kullanılarak, simetrik şifrelemeye dayanan mekanizmaları da içermek üzere, güvence altına alınabilir [38]. Özellikle, bu MAC düzlemi yaklaşımları bir paketin verici kaynağını doğrular ve bu verici kaynağının alıcıdan biraz uzakta olduğunu güvence altına alır.

3.4.5. Komşu Doğrulama

SEAD'deki her yönlendirme güncelleme mesajının kaynağı da doğrulanmak zorundadır, aksi halde, bir saldırgan yönlendirme tekrarları yaratabilir. TESLA [37], HORS [42] veya TIK [38] gibi her etkili yayın doğrulama mekanizması, komşuyu doğrulamak için kullanılabilir. Bu yaklaşımların sakıncaları senkronize saatler istemeleri ve ya bir doğrulama gecikmesi veya görece yüksek bir iletişim ek yükü gerektirmeleridir. Zaman senkronizasyonu gerektirmeyen alternatif bir yaklaşım, her düğüm çifti içerisinde paylaşılan bir gizli anahtar varsaymak ve ilgili anahtarı Mesaj Doğrulama Kodu ile birlikte kullanmaktır. Gönderici, her bir yönlendirme güncellemesi ile her bir komşu için bir Mesaj Doğrulama Kodu içerecektir. SEAD periyodik komşu tanıma fonksiyonu içerdiğinden, her bir düğüm yönlendirme güncellemelerini doğrulayacağı komşu setlerini bilir. Özellikle, her düğüm geçerli bir doğrulayıcısı olan

herhangi bir sıfır ölçülü güncellemeye güvenir. Eğer bir düğüm başka bir düğümden güncel bir dizi sayısı için bu tip bir güncelleme aldıysa, bu düğümü komşu olarak değerlendirir ve sıradaki güncellemelerde onun için bir Mesaj Doğrulama Kodu hesaplar.

İki düğüm önce komşu olursa, iki düğümden biri önce bir yönlendirme güncellemesi iletecektir. Bu güncelleme alıcı düğümün yeni komşuyu tanımasını sağlayacaktır. Bu güncellemenin duyulmasının sonucu, alıcı düğüm, diğer düğümün yeni düğümü tanımasını mümkün kılan bir tetiklenmiş yönlendirme güncellemesi gönderecektir.

3.4.6. Aynı Mesafede Hile (Same-Distance Fraud) Önleme

Ölçü ve dizi sayısını tek yönlü sağlama zinciri ile doğruluyoruz. Bu çözüm aynı mesafede hileye engel olmamaktadır: şöyle ki, dizi sayısı s ve mesafe (ölçü) d için bir ilan alan bir düğüm, aynı dizi sayısı s'i ve mesafe d'yi yeniden ilan edebilir. Aynı mesafede hileye karşı savunma için sağlama ağaç zincirleri tasarladık. Bunlar sağlama zincirlerine benzer özellikler taşırlar ancak, düşmanı kablosuz ağlardaki bir yönlendirme güncellemesini tekrarlamaktan korumak için paket kayışları [18] ile birlikte kullanıldıklarında aynı mesafede hilenin tespit edilmesini mümkün kılarlar.

Doğrulayıcıyı düğümün adresine bağlayıp bir yönlendirme ilanı göndererek, bu şekilde bir saldırganı bir komşudan duyduğu bir doğrulayıcıyı tekrar etmekten önleyerek aynı mesafede hileyi önlemekteyiz. Zincirin her elemanı düğüm kimliğini kodladığı ve bu şekilde eğer kendi kimliğini kodlamak istiyorsa bir düğümü mesafe ölçüsünü yükseltmeye zorlayan, adını sağlama ağacı zinciri koyduğumuz özel bir tek yön zinciri inşa ediyoruz. Bu tek yönlü zincirdeki her adım bir değerler topluluğu içerir, bir veya daha fazlası herhangi bir belirli düğümü doğrulamak için kullanılır. Bu yaklaşım, [43] HORS imza düzeninde kullanılana benzer. Bu değerler, bir Merkle ağaç kullanılarak doğrulanır, ve bu Merkle ağacının kökü bir sonraki adımda değerler topluluğu üretmek için kullanılır.

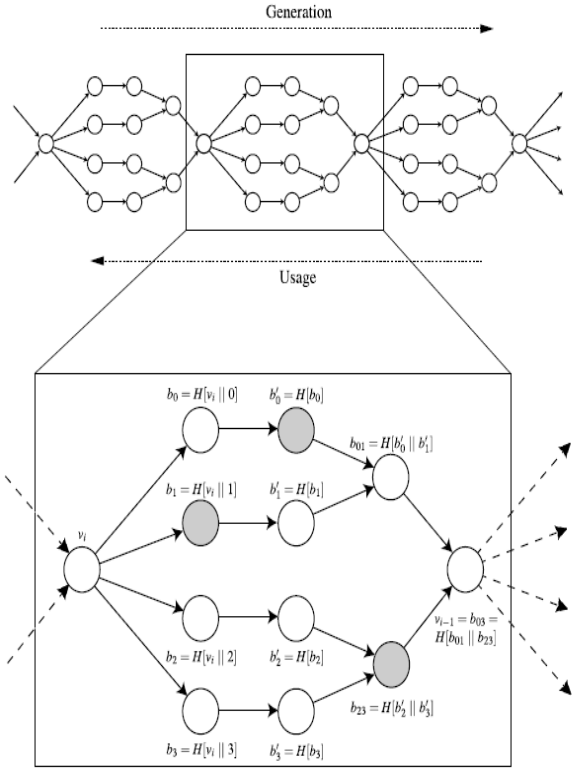
Bir sağlama ağacı zinciri bir sağlama zinciri ve tek yönlü zincir arasında bir hibriddir. Tek yönlü zincir özelliği aynı şekilde kullanılır (düğümlerin mesafe ölçüsünü azaltmamalarını zorlamak için) ve sağlama ağacı özelliği düğüm kimliğini doğrulamakta kullanılır. Sağlama ağacını tek yönlü zincir değerlerinden her vi_1; vi çifti için şu şekilde kuruyoruz: vi değerinden, tek yönlü sağlama fonksiyonu kullanarak her j için H olarak bj [HIIj] bir b0; . . . ; bn değerler seti çıkarsıyoruz. Daha sonra

doğrulama için bu değerlerin üzerinde bir sağlama ağacı kuruyoruz. Ağacın kökü, $v_{i-1} = b_{0n}$ tek yönlü zincirin önceki değeri olmaktadır. Şekil 2 bir örnek sunmaktadır. 1 kodlu düğüm, b0 0, b1, ve b23 gölgeli değerleri komşu düğümlere gönderir, bu şekilde mesafe ölçüsü kendiliğinden yükselir.

Şimdi sağlama ağacı zincirinin nasıl kullanılabileceğine dair iki örnek sunuyoruz: Tek bir değer bir düğüme denk gelirse ve değerlerin c-katı bir düğüme denk gelirse. Simgesel ve analitik kolaylık sağlamak açısından, her sağlama zinciri değeri arasındaki değerlerin sayısı ikinin kuvvetidir. Küçük bir ağda her bj değeri tek bir düğüme denk gelebilir. Hiçbir düğüm tek bir değer paylaşmadığından, bir saldırganın kendi değerini komşu düğümlerin ilanından çıkarsama yolu yoktur ve bu nedenle geçerli bir doğrulayıcı sağlamak amacıyla sağlama ağacı zincirini bir sonraki adıma takip etmelidir.

Daha büyük ağlarda, n düğümlerle, zincirin her bir adımını yaratma 0(n) ek yükü çok büyük olabilir, sonuç olarak her bir düğümü değerlerin c-katı ile doğrularız. Hiçbir iki düğüm aynı c-katı değeri paylaşmasa da, bir saldırgan kendi c değerlerinin her birini aynı ölçüyü ilan eden farklı komşulardan öğrenebilir ve ölçüyü yükseltmeden bir ilanı karıştırabilir.

Bir saldırganın başarı ihtimalinin düşük olduğunu gösterdik. Aynı zamanda her bir güncelleme için düğüm kodunun şifresini değiştiriyoruz, böylelikle bir saldırgan, komşularından uygun bir değer seti bulduğunda sabit bir ağda güncellemeleri karıştırmaya devam edemez



Şekil 3.12: Bir sağlama ağacı zincirinin bir dizisindeki ölçüm mesafesini doğrulamak. Bu örnekte, hern bi elemanı bir rotayı temsil eder, buşekilde bu sağlama ağacı zinciri dört rotayı destekler.

3.5. GÜVENLİ AODV (SAODV)

Güvenli AODV (SAODV), açık *anahtar kriptografi* bazlı AODV'nin güvenlik eklentisidir. SAODV yönlendirme mesajları (RREQler, RREPler, ve RERRler), bütünlük ve güvenirliğini garanti etmek için dijital olarak imzalanmıştır. Dolayısıyla, yönlendirme mesajı oluşturan bir devre, mesajı kendi kişisel anahtarı ile imzalar ve bu mesajı alan devreler, imzayı göndericinin kişisel anahtarını kullanarak doğrular. Atlama sayısı gönderici tarafından imzalanamaz çünkü her atlamada artmalıdır. Dolayısıyla, onu korumak için (örn. zararlı aracı devrelerin azalmasına izin vermemek için), karma zincire dayalı bir mekanizma kullanılır. Temel formunda, hedefe doğru bir yolları varsa, bu durum ara devrelerin RREQ'ye cevap vermelerini imkansız kılar, çünkü RREP mesajı hedef düğüm tarafından imzalanmalıdır. AODV'nin bu işbirliği mekanizmasını korumak için, SAODV, ara devrelerin RREQ mesajlarına cevap vermesine izin veren

bir görevlendirme özelliği içerir. Bu özelliğin adı *çift imzadır:* A düğümü, normal imzaya ek olarak bir RREQ mesajı oluşturduğunda, A'nın kendisine doğru gelen simgesel bir RREP mesajı üzerinden hesaplanmış ikinci imzayı içerebilir. Ara düğümler, düğüm A ile ilgili diğer yönlendirme bilgileri ile birlikte kendi yönlendirme çizelgeleri içinde ikinci bir imzayı kaydedebilirler. Bu düğümlerden biri A düğümüne doğru RREQ alırsa, normal AODV ile olanlara benzer bir şekilde, bir RREP mesajı ile birlikte A'nın yerine cevap verebilir. Bunu yapabilmek için, ara düğüm, daha önceden önbelleğe atılan düğümün imzasını içeren RREP mesajını oluşturur ve kendi özel anahtarı ile mesajı imzalar. SAODV, AODV ile ilgili ek bir mesaj gerektirmez. Bununla birlikte, SAODV mesajları dijital imzalardan dolayı önemli biçimde büyüktür. Buna ek olarak, SAODV önemli asimetrik kriptografik işlemler gerektirir: ne zaman bir düğüm yönlendirme mesajı oluştursa, bir imza oluşturmalıdır, ve ne zaman bir yönlendirme mesajı alırsa (aynı zamanda ara bir düğüm olarak), imzayı doğrulamalıdır. Çift imza mekanizması kullanıldığında bu durum daha da kötüleşir, çünkü tek bir mesaj için iki imzanın oluşturulması ya da doğrulanması gerekebilir. SAODV işlemlerinde, SAODV, AODV yönlendirme verisini onaylamaya izin verir. İki [19,20] mekanizma sayılanları elde etmek için kullanılır: karma zincir ve imzalar.

3.5.1. SAODV İmzalar

İmzaları hesaplarken, Atlama sayısı değişken alan olduğu için her zaman sıfırlanmıştır. RREQ Çift İmza Uzatmaya ait RREP alanı için imza durumunda, gelecek RREP mesajında imzalanan düğümler RREQ'ya karşılık olarak geri gönderilebilirler. Bu mesajı yapılandırmak için, RREQ ve Önek Boyutu değerlerini kullanır (RREQ'den türetilebilir olmayan ve imza hesaplarken sıfırlanmamış RREP alanı. RREPlerin olması durumunda, R ve A işaretleri de sıfırlanmıştır. SAODV, AODV çoklu gönderimini dikkate alarak tasarlanmamıştır ('R' işareti çoklu gönderimde kullanılmıştır), 'A' işareti değişkendir ve herhangi bir saldırgan onu değiştirirse, sadece hizmet ile ilgili bir engel durumuna sebep olabilir. Bir düğüm RREQ oluşturduğunda, Tek İmza Uzatma ya da Çift İmza Uzatma ile imzalanıp imzalanmaması gerektiğine karar verir. Tüm uygulamalar RREQ Tek İmza Uzatmayı DESTEKLEMEK ZORUNDADIR ve RREQ Çift İmza Uzatmayı DESTEKLEMELİDİR. Karşılıksız RREP işareti ile birlikte bir RREQ oluşturan düğüm, RREQ'yi Çift İmza Uzatma ile imzalamalıdır. Bir düğüm İmza Uzatma eklemeden hiçbir zaman RREQ oluşturmamalıdır.

Bir düğüm RREQ aldığında, sunucu için özel bir yol oluşturmadan ya da güncellemeden önce imzayı doğrular. Sadece imza doğrulandığında, yolu belleğinde saklayacaktır. RREQ Çift İmza Uzatma ile alınırsa, o zaman düğüm de imzayı, süreyi ve yol girişindeki RREQ için olan Hedef IP adresini saklayacaktır. Bir düğüm İmza Uzatımı olmadan RREQ alırsa, onu BIRAKMALIDIR. Sadece eğer AODV gerekliliklerini yerine getirirse, ara bir düğüm RREP ile birlikte RREQ'ya yanıt verecektir ve RREP Çift İmza Uzatımının 'İmza', 'Eski Süre' ve 'Eski Kaynak IP adresi' alanlarına koymak için devrenin uygun imza, eski süre ve eski kaynak IP adresi bulunmaktadır. Aksi takdirde, RREQ'yi tekrar yayınlayacaktır. RREQ hedef tarafından alındığında, sadece AODV gerekliliklerini yerine getirirse RREP ile cevap verecektir. Bu RREP, bir RREP Tek İmza Uzatımı ile gönderilecektir.

Tüm uygulamalar RREP Tek İmza Uzatımını DESTEKLEMEK ZORUNDADIR ve RREP Çift İmza Uzatımını DESTEKLEMELİDİR. Bir düğüm asla İmza Uzatımı eklemeden bir RREP oluşturmamalıdır. Bu durum aynı zamanda karşılıksız RREPler için de geçerlidir. Bir düğüm RREQ aldığında, sunucu için özel bir yol oluşturmadan ya da güncellemeden önce imzayı doğrular. Sadece imza doğrulandığında, imza, süre ve RREP'ye ait kaynak IP adresi ile birlikte yolu saklayacaktır. Bir düğüm RREP'yi Tek Uzatma olmadan alırsa, BIRAKMALIDIR. RERR mesajı oluşturan ya da gönderen her düğüm, tüm mesajı imzalamak için dijital imzalar kullanır ve bu mesajları alan komşular imzayı doğrular. Bu yolla, RERR mesajının göndericisinin kimliğini doğrulayabilir. Ve, hedef sıra numaraları uygun düğümler tarafından imzalanmadığı için, bir düğüm RRER mesajı baz alınarak yönlendirme tablosunun herhangi bir hedef sıra numarasını güncellememelidir. Düğümler, RERR mesajı içinde bulunan hedef sıra numaralarına güvenmeseler dahi, bir yolu iptal edip etmemelerine karar verme konusunda kullanacaklardır.

3.5.2. SAODV Karama Zincirler

SAODV, her düğümün ara düğümü de içeren mesajı almasına izin verir bir şekilde RREQ ve RREP mesajlarının atlama sayısını ve atlama sayısının herhangi bir saldırgan tarafından eksilmediğini doğrulamak için karma zincir kullanır. Karma zincir, kaynağa aralıksız olarak tek yollu karma işlevi uygulayarak oluşturur. Düğüm RREQ ya da RREP mesajı oluşturduğundaki işlemler aşağıdaki gibidir:

1. *Kaynak* olarak rastgele bir sayı oluşturur ve değeri kaynak olarak kullanmak

için *Karmayı* belirler

Karma = kaynak

2. IP başlığından TimeToLive (Süre) içinde dosyalanmak için Max_Hop_Count (maksimum atlama sayısını) belirler

Max_Hop_Count (maksimum atlama sayısı) = TimeToLive (süre)

3. Hangi karma işlevinin uygulandığını belirtmek için *Hash_Function (karma işlevini)* belirler.

Hash_Function (Karma işlevi) = h

4. Max_Hop_Count (maksimum atlama sayısı) kadar kaynak değeri hesaba dayalı adreslemek için Top_Hash (üst karmayı) hesaplar.

Top_Hash (üst karma) = h *Max_Hop_Count (maksimum atlama sayısı)– Hop_Count (atlama sayısı)*(*Hash*) *(Karma)*

h, karma işlevi anlamına gelir ve $h^i(x)$, *i* kez *h*-*x* işlevini uygulama sonucudur.

Atlama sayısını doğrulamak ve RREQ ya da RREP mesajlarını tekrar iletmek için, düğüm aşağıdaki işlemleri uygular:

1. Karma alanı içindeki değer kaynağını bulmak için *Max_Hop_Count'tan (maksimum atlama sayısı) Hop_Count'u (atlama sayısı)*çıkararak *Hash_Function (Karma işlev)* alanı tarafından belirlenen karma işlevini uygular ve değerin Top_Hast (üst karma) alanı içinde bulunan değere eşit olduğunu doğrular.

Top_Hash (üst karma) = hMax_Hop_Count (maksimum atlama sayısı) – Hop_Count (atlama sayısı) (Hash) (Karma)

2. RREQ'yı tekrar yayınlamadan ya da RREP'yi göndermeden önce, yeni atlama için Karma değeri ile ilgili karma işlemi amacıyla bir düğüm uygulanır.

Hash (Karma) = h(*Hash*) (Karma)

Hash_Function (Karma işlevi), Max_Hop_Count (maksimum atlama sayısı), Top_Hash (üst karma), ve Hash (Karma) alanları dijital imza ile iletildiği için,

3.5.3. SAODV Özellikleri Aşağıdaki Gibidir:

i. Onaylı açık anahtarların mülkiyeti, ara düğümlerin tüm iletilen yönlendirme paketlerini doğrulamasına izin verir.

ii. Protokol genel olarak, AODV protokolü ile yeni uzatma mesajları kullanarak işlem görür. iii. SAODV, bütünlük, doğrulama ve reddedilmeme gibi güvenlik özellikleri sağlayarak AODV yol keşif mekanizmasını korumak için kullanılabilir.

Sayılanlar daha sonra açıklanacaktır, *Hash (karma)* alan dışındaki hepsi bütünlüğü korumak için imzalanacaktır.

3.5.4. SADOV Dijital İmzalar

Dijital imzalar RREQ ve RREP mesajlarında bulunan değişmez veri bütünlüğünü korumak için kullanılır. Bunun anlamı, AODV mesajının *Hop_Count (atlama sayısı)* ve SAODV uzatımından gelen *Hash (karma)* dışında her şeyi imzalarlar.

Dijital imzalar için başvuruda bulunmanın ana problemi, AODV'nin hedef için 'yeteri kadar taze' yolları varsa, ara düğümlerin RREQ mesajlarına cevap vermesine izin vermesidir. Geçici Mobil Ağ Kurma (MANET) tarafından önerilen çözüm, iki alternatif olduğu yönündedir: İlk alternatif, sadece final hedeflerin RREQ'ya cevap vermesine izin veren SAODV İmza Uzatımıdır, ikinci ise belirli bir sınırlama olan SAODV Çift İmza Uzatımıdır.

3.5.5. SAODV İmza Uzatımı

Bir RREQ gönderildiğinde, gönderici kendi özel anahtarını kullanarak AODV paketi düzenlenerek oluşturulan bir imza hazırlar. Ara düğümler sunucu için ters bir yol oluşturulmadan ya da güncellenmeden önce imzayı onaylar ve eğer sadece imza onaylanmış ise yolu kaydederler. Son hedef düğümü özel anahtarı ile birlikte RREP'yi imzalar. Ara ve son düğümler de tekrar kullanmadan önce imzayı doğrular.

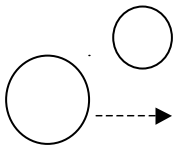
3.5.6. SAODV Çift İmza Uzatımı

Bir RREQ gönderildiğinde, gönderici mesajı kendi özel anahtarı ile imzalar. Ara düğümler sunucu için ters bir yol oluşturulmadan ya da güncellenmeden önce imzayı onaylar ve eğer sadece imza onaylanmış ise yolu kaydederler. Fakat aradaki fark, eğer RREP Çift İmza Uzatımı ile alınmış ise, o zaman, düğüm yol girişi içinde RREP ve süre için imzayı saklar. Ara bir düğüm RREQ'ye cevap vermek isterse, düğümün sadece doğru yolu değil aynı zamanda uygun imzası da olmalıdır. O zaman, kaydedilen imza

ve süreyi ekleyerek Çift İmza Uzatımı ile birlikte RREP oluşturur ve gerçek süreyi imzalar ve gönderir. RREP alan ve yolu güncelleyen tüm düğümler, doğrulama işleminden sonra yol ile birlikte imza ve süreyi saklarlar.

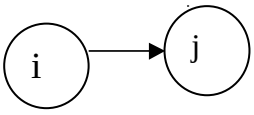
3.6. FAODV

Kara deliğin iki özelliği vardır.[54][55][56] İlk olarak paketleri durdurma niyeti ile yol sahte olsa dahi, kendini varış düğümüne doğru geçerli bir yol sahibi olarak tanıtmak için AODV gibi geçici yönlendirme protokollerinden yararlanır. İkinci olarak, düğüm önü kesilen paketleri tüketir. Protokol temsillerinin aşağıdaki anlaşmalarını tanımlarız.

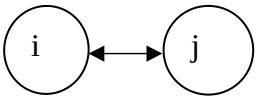


Düğüm : Mobil Birim

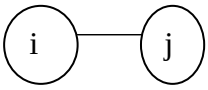
Kesikli Ok: RREQ/RREP Yayılma



Düz Ok (Tek başlı) : Düğüm i'nin düğüm *j'ye* doğru bir yolu vardır düğüm j'ye doğru gönderilen veri paketleri bulunmaktadır (j düğüm i için güvenilir bir düğümdür)



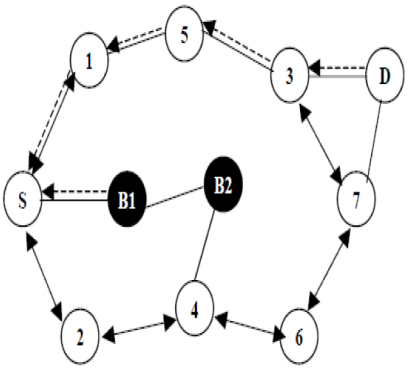
Düz Ok (Çift başlı): *i* ve *j* düğümlerinin birbirlerine doğru yolları ve gönderilen veri paketleri vardır (i ve j birbirleri için güvenilirdir)



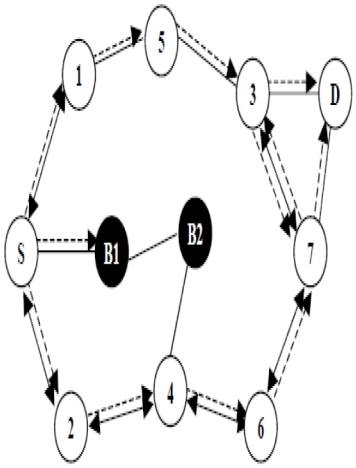
Düz Çizgi: Her iki düğüm arasında bir yol vardır

3.6.1. Kara Delik Saldırısı

Orijinal AODV protokolüne göre, kaynak düğüm S hedef düğüm D ile iletişim kurmak istediğinde, kaynak düğüm S (RREQ) paketine yol talebi gönderir. Aktif komşu düğümler kaynak düğüm S girişi ile birlikte yol tablolarını günceller ve hedef düğümüne yeterli kadar yol gidip gitmediğini kontrol ederler. Gitmiyorsa, ara düğüm RREQ'yu günceller (atlama sayısını arttırarak) ve düğüm D'ye ya da Örnek 1'deki örnekte tarif edildiği şekilde D'ye yeteri kadar yolu olan diğer ara düğümlere ulaşana kadar hedef düğüme giden RREQ ile ağı gönderi yapar. D'ye yeteri kadar yolu olan hedef düğüm D ya da ara düğüm, şekil 3'te tarif edildiği şekilde yer yöndeki yol yanıtını (RREP) başlatır. Düğüm S, ilk olarak cevap veren komşu düğümlere veri paketleri gönderir ve diğer cevapları gözden çıkarır. Ağda zararlı düğüm olmadığında bu işlem başarılı bir şekilde çalışır.



Şekil 3.13: RREP mesajlarının yayılması.



Şekil 3.14: RREQ'nin ağ akışı.

Araştırmacılar tek kara deliği tanımlamak ve elemek için çözümler sunmuşlardır [53]. Ancak, koordinasyon içinde hareket eden çoklu kara deliklerin durumuna değinilmemiştir. Örneğin, çoklu kara delik düğümleri birbirleri ile koordinasyon içinde hareket ederlerse, ilk kara delik düğümü B1, Şekil 3.13'de tarif edilen bir sonraki atlama sayısı gibi takım arkadaşları B2'den birisini. [53], kaynak düğüm B1 yolu dışında farklı bir yolla (S-2-4-B2) B2'ye "Sonraki Talep (FRq)" gönderir. Düğüm S, B2'ye düğüm

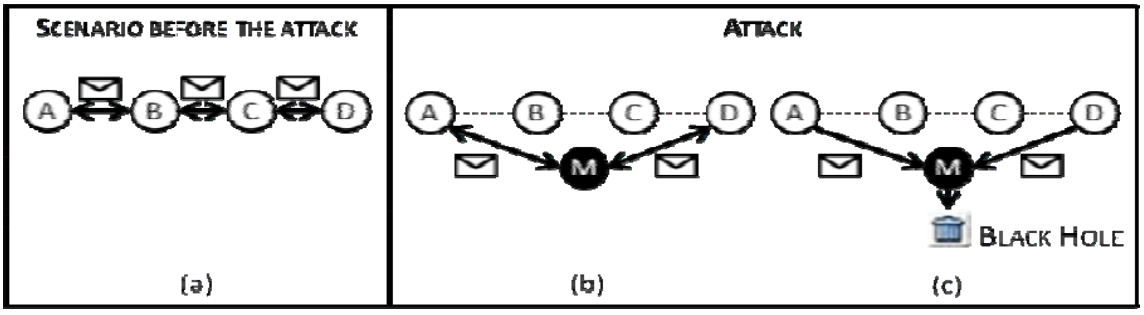
B1'e varış düğümü D'ye yolu olup olmadığını sorar. Çünkü B2, B1 ile işbirliği içindedir, "Sonraki Talebi (FRq)" her iki soruya da "evet" yanıtı olacaktır. Düğüm S, yol S-B1-B2'nin güvenli olduğunu varsayarak, veri paketlerini geçirmeye başlar. Ancak, gerçekte paketler düğüm B1 tarafından tüketilir ve ağın güvenliği ihlal edilir.

Yönlendirme protokollerine dönük tehditlerin iki kaynağı vardır. İlki dış saldırganlardan gelir. Bir saldırgan, hatalı yönlendirme bilgileri aşılayarak, eski yönlendirme bilgilerini tekrar ederek veya yönlendirme bilgilerini bozarak bir ağı başarıyla bölümlendirebilir veya yeniden iletim ve verimsiz yönlendirme yaratarak bir aşırı trafik yüklenmesi ortaya koyabilir. Tehdidin ikinci ve daha şiddetli türü gizliliği ihlal edilmiş düğümlerden kaynaklanabilir. Bunlar (i) diğer düğümlere yönlendirme bilgisini kötüye kullanabilir, (ii) hizmet hatalarına yol açmak amacıyla uygulamalı verileri temel alarak hareket edebilir.

Belirli yönlendirme protokolleri üzerindeki bu tür tehditlerin etkisini değerlendirmek için sistematik yaklaşımlar sağlamak bugün açık bir sorun olmaya devam etmektedir.

Söz konusu saldırı yaklaşımı bahsi geçen zorlu işle başa çıkabilir. Birbirini izleyen iki adımda yapılandırılır (bkz. Şekil 3.15):

1. Kötü niyetli düğüm (M) başarılı bir saldırı için (Şekil 3.15.b) elverişli bir ağ topolojisine neden olur. Bu amaçla başa çıkabilmek için (i) M saldırının hedefi olan cihazlar arasında (A ve D diyelim) bir yönlendirme bağına sebep olur sonra (ii) M, A ve D'yi iletişimleri için bu tip bir bağ seçmeye yönlendirmek için protokol uyumlu mesajlar gönderir.
2. M saldırıyı yürütmektedir (Şekil 3.15.c.). Bir kara delik saldırısı durumunda M paketleri bırakır (yeniden aktarmaz). Bu paket sızıntısı seçici (sadece belirli bir tipteki paketleri etkiler) olabilir veya olmayabilir (tüm paketler kara deliklidir).



Şekil 3.15: Saldırı yaklaşımı.

Diğer tip saldırılarla ilgili ayrıntılı bir açıklama bu makalenin kapsamı dışındadır. Ancak, M'nin yolu kesilen paketleri yönlendirme biçimini basitçe değiştirerek başka türlü saldırılar gerçekleştirebileceğini belirtmekte fayda vardır (bkz. Şekil 3.15.c.). Örneğin, yeni paketler oluşturabilir veya yolu kesilmiş paketleri değiştirebilir, geciktirebilir veya yeniden çağırabilir.

Saldırı bir kez enjekte edildikten sonra, bir ağ iletişimi ve çalışan uygulamalardaki etkisi değerlendirilmelidir. Bu etki, örneğin, belirli bir uygulamanın başarısız olmasına, ağ iletişiminin çözülmesine, düğümlerin izole edilmesine veya yönlendirme döngüleri oluşturmasına yol açabilir

3.6.2. Çözüm

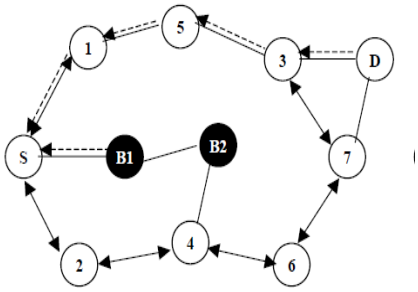
Bu bölümde, Veri Yönlendirme Bilgisi (DRI) Tablosu ve Çapraz Kontrol tanıtarak az oranda FAODV bir protokol sahibi bir grup olarak işbirliği içinde bulunan çoklu kara delik düğümlerini tanımlamak için bir yöntem bilim teklifinde bulunuruz.

3.6.2.1. Veri Yönlendirme Bilgi Tablosu

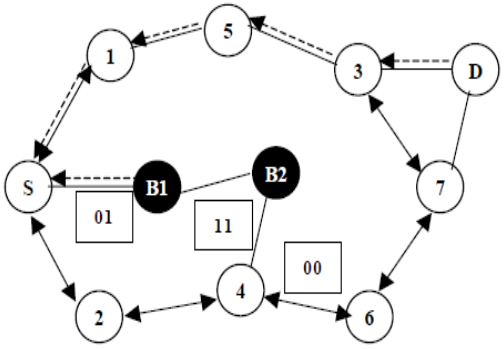
İşbirliği içinde hareket eden çoklu kara delik düğümlerini tespit etmek için çözüm, kaynak düğüm S'nin RREQ'suna cevap veren düğümlerden iki bitlik ek bilgi içerir. Her düğüm ek bir Veri Yönlendirme Bilgisi (DRI) tablosu muhafaza eder. DRI tablosunda, 1'in anlamı 'doğru' ve 0'ın anlamı 'yanlıştır'. İlk bit "Gelen" düğümden *gelen* yönlendirme veri paketi ile ilgili bilgiler anlamına gelir (Düğüm alanı) ve ikinci bit "Aracılığıyla" düğüm *aracılığıyla* yönlendirme veri paketi ile ilgili bilgiler anlamına gelir (Düğüm alanı içinde). Örnek 3'e bir örnek olarak, düğüm 4 ile korunan bir veritabanı örneği Tablo 3.5'de gösterilmiştir. Düğüm için olan 1 0 girişi, düğüm 4'ün 3'ten veri paketleri gönderdiği fakat 3 aracılığıyla veri paketi göndermediği anlamına gelir (düğüm 3, 4'ten uzaklaşmadan önce). Düğüm 6 için olan 1 1 girişi, düğüm 4'ün başarılı bir şekilde düğüm 6'dan ve aracılığıyla veri paketi gönderdiği anlamına gelir. Düğüm B2 için olan 0 0 girişi, düğüm 4'ün B2'den ya da aracılığıyla herhangi bir veri GÖNDERMEDİĞİ anlamına gelir.

Tablo 3.5: Ek veri tablosu düğüm 4'ten gelen ve korunan veri tablosuna gönderilmiştir.

Düğüm #	Veri Gönderme Bilgisi	
	Gelen	Aracılığıyla
3	1	0
6	1	1
B2	0	0
2	1	1



Şekil 3.16: Kara delik saldırısını önlemenin yolları.

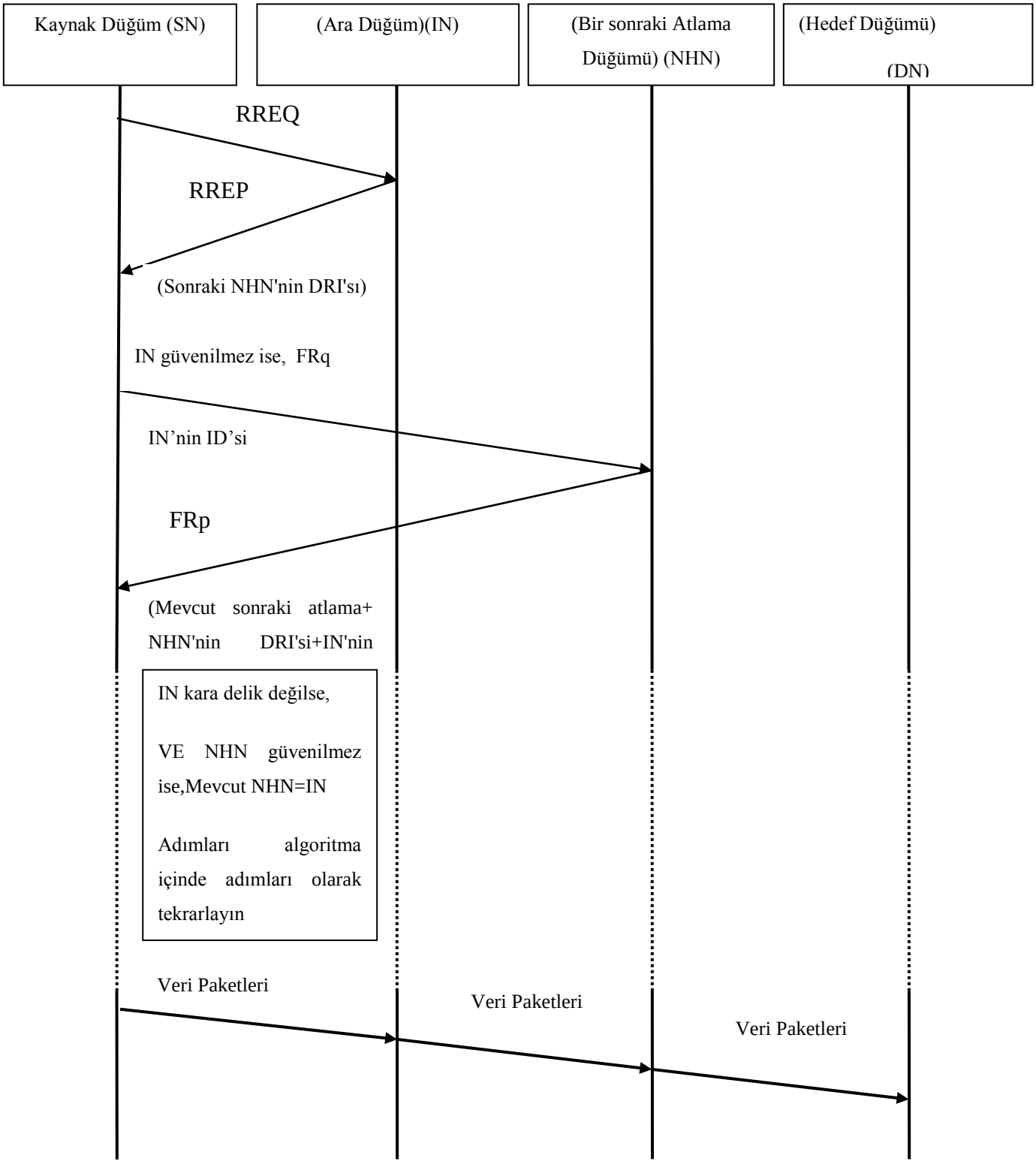


Şekil 3.17: Tek bir kontrol ile kara Delik düğümlerini tanımlamanın yolları.

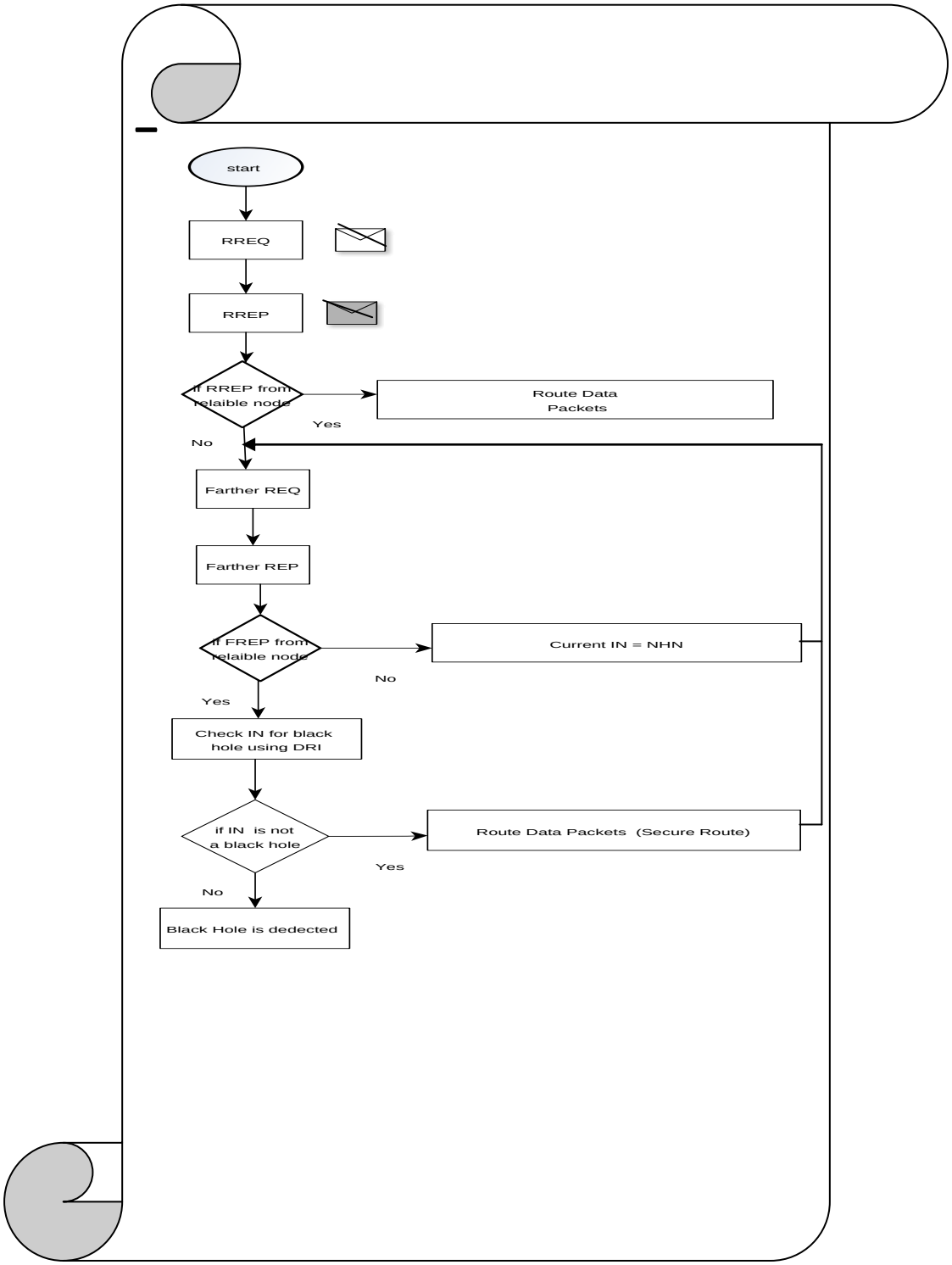
3.6.2.2. Kntrol Kıyaslaması

Tekniklerimizde, güvenilir düğümlerin (kaynak düğümün veri gönderdiği düğümler) veri paketi göndermesine güveniriz. FAODV protokolü ve teklif edilmiş yöntem bilimimiz için olan algoritma Şekil 3.18 ve 3.19'te gösterilmiştir. Protokol içinde, kaynak düğüm (SN) varış düğümüne giden güvenli yolu keşfetmek için bir RREQ mesajı yayınlar. RREP oluşturan Ara Düğüm (IN), Sonraki Atlama Düğümünü (NHN) ve NHN için olan DRI girişini sağlamak zorundadır. IN'den gelen RREP mesajının üzerine, kaynak düğüm, IN'nin güvenilir bir düğüm olup olmadığını görmek için kendi DRI tablosunu kontrol edecektir. Kaynak düğüm yol verisinden önce IN kullanırsa, o zaman IN güvenilir bir düğümdür ve kaynak düğüm IN aracılığıyla yönlendirme verisini başlatır. Aksi takdirde, IN güvenilmezdir ve kaynak düğümü, IN'nin tanımlanmasını kontrol etmek ve NHN'ye sormak için NHN'ye bir FRq mesajı gönderir: 1) IN'nin NHN aracılığıyla gönderilmiş verisi varsa, 2) mevcut NHN'nin varışa bir sonraki atlamasının kimliği 3) bir sonraki atlama aracılıyla mevcut NHN

gönderilmiş veriye sahip olup olmadığı. NHN buna karşılık olarak FRp mesajı ile cevap verir, mesajın içerdikleri 1)IN için DRI girişi, 2) mevcut NHN'nin sonraki atlama düğümü, ve 3) mevcut NHN'nin sonraki atlaması için giriş. NHN'den alınan FRp mesajı baz alınarak, kaynak düğümü NHN'nin güvenilir olup olmadığını kontrol eder. Kaynak düğümün önceden NHN aracılığıyla gönderilmiş bir verisi varsa, NHN güvenilirdir; aksi takdirde, güvenilir değildir. NHN güvenilir ise, kaynak düğüm IN'nin kara delik olup olmadığını kontrol eder. IN'den gelen DRI girişinin ikinci biti (örn. IN, NHN *aracılığıyla* veri gönderdiyse) 1'e eşit ise, ve NHN'den gelen DRI girişinin ilk biti (örn. NHN, IN *aracılığıyla* veri gönderdiyse) 0'a eşit ise, IN bir kara deliktir. IN kara delik değilse ve NHN güvenilir bir düğümse, yol güvenlidir ve kaynak düğüm 01 ile IN için DRI girişini güncelleyecektir ve IN yoluyla veri göndermeye başlar. IN kara delik ise, IN'den RREP oluşturan düğüme kadar ters yol boyunca tüm devreleri kara delik düğümleri olarak tanımlar. Kaynak düğüm kara deliklerden gelen diğer RREP'leri görmezden gelir ve işbirliği yapan kara deliklerin listesini yayınlar. NHN güvensiz bir düğüm ise, kaynak düğüm mevcut NHN'ye IN olarak işleme alır, güncellenmiş IN'nin bir sonraki atlama düğümüne FRq gönderir ve algoritma içinde olan adımlarla bir döngü içinde gider.



Şekil 3.18: İşbirliği yapan kara delik saldırısını engellemek için FAODV protokol ve algoritması.



Şekil 3.19: FAODV protokolu kara delik saldırısını engellemek algoritması .

Örneğin, Şekil 3.17'teki ağı düşünelim. Düğüm B1, RREP mesajı ile birlikte kaynak S'ye cevap verirse, sonraki atlama için sonraki atlama düğümü B2 ve DRI'yi sağlar (örn. B1, B2 yoluyla veri paketleri gönderdiyse). Bu örnekte kara delik düğümü, 0.1'e eşit DRI değeri ile cevap vererek yolu kullanma ile ilgili yanlış bilgi vermektedir.

B1'den RREP mesajının alınması üzerine, kaynak düğüm S, B1'in güvenilir bir düğüm olup olmadığını görmek için kendi DRI tablosunu kontrol edecektir. S daha önce B1 aracılığıyla hiç veri göndermediği için, B1, S için güvenilir bir düğüm değildir. Daha sonra, S alternatif yol S-2-4-B2 B2'ye FRq gönderir ve B2'ye, B2'nin sonraki atlaması olan B1'ye veri gönderip göndermediğini ve B2'nin B2'nin sonraki atlaması aracılığıyla veri paketi gönderip göndermediğini sorar. B2, B1 ile birlikte çalıştığı için, tüm üç talebe pozitif olarak cevap verir ve sonraki atlamada düğüm 6'yı sağlar (rastgele). Kaynak düğüm, düğüm B2 iddialarını çapraz kontrol etmek için alternatif yol S-2-4-6 yoluyla düğüm 6 ile temasa geçtiğinde, düğüm 6 negatif cevap verir. Düğüm 6'nin düğüm B2'ye giden bir yolu yoktur ya da düğüm B'den veri paketi almamıştır, B2'ye uyumlu DRI değeri, Şekil 4'te gösterildiği şekilde 0 0'a eşittir. Bu bilgi baz alınarak, düğüm S, B2'nin kara delik düğümü olduğu anlamına gelebilir. Düğüm B1'in, düğüm B2 aracılığıyla veri paketleri göndermesi gerekiyorsa, göndermeden önce düğümü doğrulamalıdır. Şimdi, düğüm B2 düğüm 6 aracılığıyla geçersiz kılındığı için, düğüm B1, düğüm B2 ile birlikte çalışmalıdır. Bu yüzden, düğüm B1 ve B2 kara delik düğümü olarak işaretlenmiştir ve bu bilgi ağları boyunca kayıtlarına kara delikler olarak yayılmıştır. Ayrıca, S, B1 ve B2'den gelen diğer yanıtları göz ardı eder ve D'ye geçerli ve alternatif bir yol arar.

Ara düğümlerin çapraz kontrol süreci, çoklu kara delik düğümlerinden ağ güvenliği sağlamak için bütçeye uygun olduğunu düşündüğümüz bir kerelik bir işlemdir. Düğümlerin çapraz kontrol maliyeti, düğümlerin birbirleriyle güvenilir düğüm listelerini (DPI tablosu) paylaşmalarına izin verilerek azaltılabilir.

3.7. DOĞRULAMA ALGORİTMASI

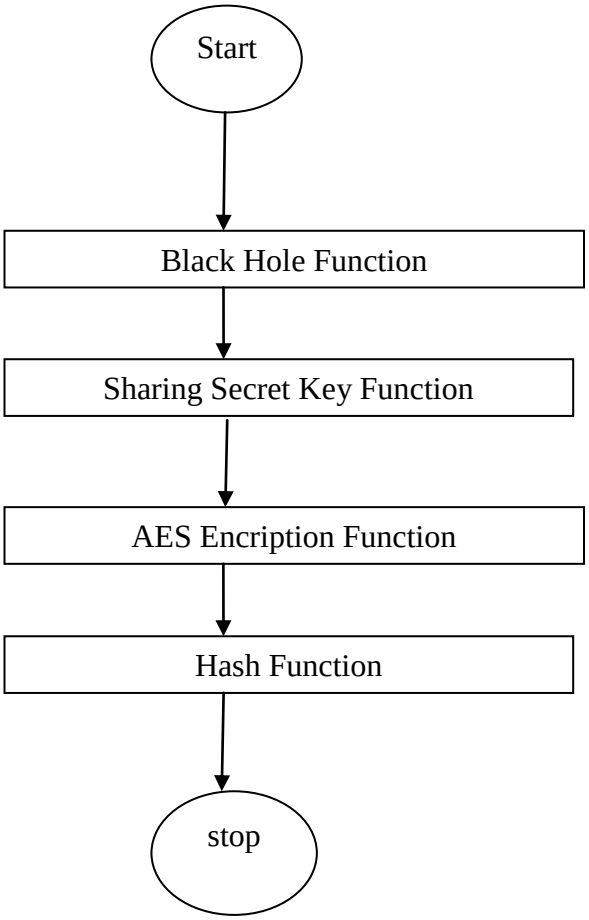
Tasarsız doğrılama doğrulaması, sürecinden bağımsızdır. 802.11 standardı, çeşitli grop doğrulama ve anahtar yönetimi iletişim kurallarına bir çerçeve sağlar. Her biri kimlik doğrulamaya farklı bir yaklaşım getiren ama hepsi istemci ile erişim noktası arasındaki iletişim için aynı 802.11 iletişim kuralı ile çerçevesini kullanan değişik 802.1X kimlik doğrulama tipleri vardır. [55][56][57][58][59][60][61][62][63] Çoğu iletişim kuralında.

Kriptografik özet fonksiyonu çeşitli güvenlik özelliklerini sağlayan bir özet fonksiyonudur. Veriyi belirli uzunlukta bir bit dizisine, (kriptografik) özet değerine, dönüştürür. Bu dönüşüm öyle olmalıdır ki verideki herhangi bir değişiklik özet değerini değiştirmelidir. Özetlenecek veri mesaj, özet değeri ise mesaj özeti veya kısaca özetolarak da adlandırılır.

İdeal bir kriptografik özet fonksiyonu şu dört özelliği sağlamalıdır:

- Herhangi bir mesaj için özet hesaplamak kolay olmalıdır.
- Bir özete karşılık gelecek mesajı oluşturmak zor olmalıdır.
- Özeti değişmeyecek şekilde mesajı değiştirmek zor olmalıdır.
- Aynı özete sahip iki farklı mesaj bulmak zor olmalıdır.

Kriptografik özet fonksiyonları, bilgi güvenliği konuları olan sayısal imza, mesaj doğrulama kodu ve diğer doğrulamayöntemlerinde yaygın olarak kullanılmaktadır. Sıradan özet fonksiyonları gibi veriyi komut çizelgesine eşlemede, eşdeğer veri bulmada, dosyaları tekil olarak tanımlamada, ve veri bütünlüğü sağlamasında da kullanılır. Bilgi güvenliği konularında kriptografik özet fonksiyonları, terim anlamları farklı olsa da sayısal parmak izi, sağlama, veya özet değeri ile benzer anlamlarda kullanılır



Şekil 3.20: FAODV protokolu doğrulama fonksiyon sınıflandırılması.

- Düz metin M orjınal mesaj, düğüm vasıtasıyla gönderilmiştir ===> RREP

- Şifre metin C_M Şifre sistem ile Şifrelenmiş M mesajı Verimidir(AES).

- Zaman senkronizasyonu: iletişim ağa dayalıdır, ona göre her düğüm aynı zaman senkronize eder. Düğüm Bölümleri bütün zamanı eşit aralıklara böler.

- Gizli anahtar paylaşımı:

  Anahtar üretimi:

  p,q seç. P ve q asal sayı p≠ q

  n=p*q de.

  Ø(n) = (p-1) (q-1) hesapla.

  Tamsayı e seç

  gcd(Ø (n)e)=1 1<e< Ø (n)

  'd' hesapla?

  d= e' mod Ø(n)

  Genel Anahtar KU= (e,n)

  Özel anahtar KR= (d,n)

- $M=D_{Ki}(E_{Ki}(M))$
- Her düğüm simetrik şifreleme sistemi taşımaktadır D_{ki} şifreleme çözme sistemi ve E_{ki} şifrelemelanmiş
- Mesaj Doğrulama kodu MAC(K_i,M).

Doğrulama Mekanizmasi

- Düğüm gizli anahtar paylaşıyor K_i
- Sistem zamanına göre düğüm zamman dilimi (Ts) üretiyor
- M ve Ts girildi düğümü olarak uygulandığımızda
  Hash Fonksiyonu $MAC_M=H(k_i,T_s,M)$

  Ek_i (MAC_M,T_s ,M)

  Simetrik evrensel şifreleme sistemi düğüm D uygulandığında

  PKT={ $Ek_i(T_s,M,MAC_M)$}

  S düğümü verim paketi oluşturup gönderiyor

  MAC_T =H(K_{iT}) zamanı uygun zamanıdır

  GF(p) Galois fieldidir ve $a_1,a_2,$……..a_{t-1} Gf(p)

  Bir a_{t-1} formüle göre seçilmektedir

  $f(x)$=KR+ $a_1x+a_2x^2+$.....$a_{t-1}x^{t-1}$ mod N.....(3,1)

  Denklemde x değerine uygulandığında

$y_i = f(v_i) \bmod N$ (i=1,2,.....N) değeri alınmakttadır.

Global simetrik kroptosistemi uyglanıp şifreleme paketi alınıp gönderilir

$$f(x)=\sum_{i=1}^{t} yi \prod_{j\neq 1} (X - vj)/(vi - vj) \bmod N \quad \text{.....(3,2)}$$

X=0 ise

$$f(x)= \text{KR} = \sum_{i=1}^{t} yi \prod_{j\neq 1} (vj)/(vi - vj) \bmod N \quad \text{.....(3,3)}$$

$$Ki = yi \prod_{j\neq i,\, j\neq 1} (vj)/(vi - vj) \bmod N \quad \text{.....,....(3,4)}$$

Global simetrik kroptosistemi uyglanıp şifreleme paketi alınıp gönderilir.

Şart 1 $P = E_{ki}(Ts,M,MAC_M)$

Şart2 $Mac_M=H,(k_i,T_s,M)$

Şart3 T_s gecikme zaman dilimi

Şart4 T_s şifreyi çöyüyor paketşnde olduğu gibi

4 Şartı gerçekleştiğinde

paket çalışır .

yoksa paketten vazgeçilecek ve yeni sinyal alternatif yol bulmak için oluşturuyor.

3.8. DENEYSEL KURULUM

Senaryoya dayalı deneylerde, açık kaynak dağıtımı olarak kullanılabilir ns-2 simülatörü kullanırız [30]. Özellikle, ns-2.35 sürümü ubuntu 12.10 çevresi üzerinde kullanılmıştır. Senaryolar oluşturmak için, hareketlilik senaryosu oluşturma aracı olan *BonnMotion*'ı

kullanırız. İki ışınlı yer yansıtma örneğine dayalı ns-2 simülatöründe CMU'ya ait kablosuz uzantıları değerlendirdik. Radyo örneği, 2 Mbps maksimum hava bağlantısı oranında çalışan 802.11 WaveLAN'a uyumludur. Ortam Erişim Kontrol protokolü IEEE 802.11 Dağıtılmış Koordinasyon İşlevini (DCF) kullanır. Trafik modeli dosyası, standart ns-2 dağıtımı ile birlikte sağlanan "cbrgen.tcl" senaryosu kullanarak oluşturulmuştur. Simülasyonlarımız için aşağıdaki parametrelerle birlikte CBR trafiğini kullanırız.

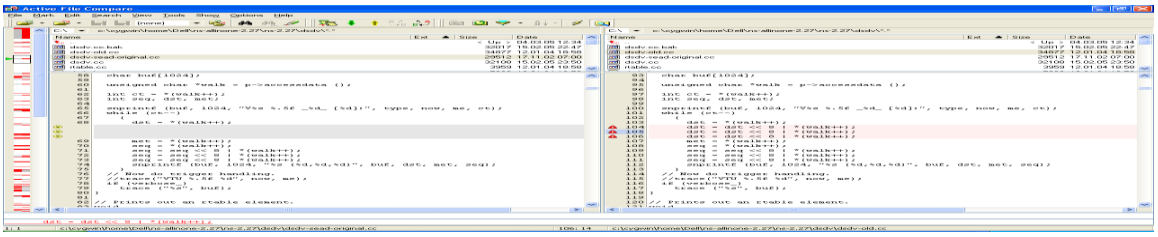
Tablo 3.6: Senaryolar için trafik örneği.

Trafik Örneği	
Bağlantıların maksimum sayısı	20
Uygulama verisi yük boyutu	512 bayt
Paket oranı	4 paket / saniye

Dolayısıyla, savaş yeri operasyonları için ses ve veri ileten yüksek hareketli radyolardan oluşan kendi kendine şekil alan ağlar olan Combat Network Radio (CNR) gibi uygulamalara uygun bir 16 Kbps bant genişliği kullanılmıştır. Deneylerin nasıl yürütüldüğü ile ilgili daha fazla detaylı açıklama için, lütfen ekte verilen öğretici bilgilere bakın.

3.9. KARŞILAŞILAN SORUNLAR

Bu bölüm deneyleri yürütürken ve sorunları çözerken karşılaşılan bazı problemleri listeler.



Şekil 3.21: Aktif Dosya Karşılaştırılması Ekran Görüntüsü.

3.9.1. Java esaslı izleme dosyası Derleyici Hatası

Çıkış dosyalarını izlemek ve tüm devrelerde, verilen senaryo ve yönlendirme protokolü için alınan dosyaların toplam sayıları, paket teslim bölümü, ortalama uçtan uca gecikme

vs. gibi istatistikleri oluşturmak için bir Java programı kullanılır. Grafik elde etmek için Excel hesap tablosu içine aktarılan virgül ile ayrılmış (.csv) dosya çıkışını oluşturur.
Yeni kablosuz izleme dosyası formatı ile sağlanan öğretici belgelerde bulunabilir. Programdaki hatalardan biri, alınan paketlerin toplam sayısını ve ağ içindeki uçtan uca gecikmeyi hesaplayan aşağıdaki kod bölümünü kullanarak yayılan gecikme hesabıdır-

```
        // alınan paket sayısını ve uçtan uca gecikmeyi hesaplayın
if (tokens[0].equals("r") && tokens[18].equals("AGT") && tokens[34].equals("cbr"))
{
                receives++;
                end_time[packet_id] = time;
 }
```

Öncelikle koyu kırmızı olarak gösterilen şartlar; örn. simgeler [18].eşitlikler ("AGT"), dahil edilmemiştir. Bu durum, programın CRB tipi trafik olan *tüm paketleri* hesaplamasına yok açar. Buna aynı zamanda ağ içindeki yönlendirme paketleri de dahildir. Bu sebeple, elde edilen uçtan uca gecikme mevcut uçtan uca gecikmeden çok daha yüksektir. Bu durum düzeltildiğinde, doğru değerler elde edilir.

3.10. SENARYOLAR

Bu deneyler için kullanılan ölçümler, bir darboğaz senaryosu içinde AODV ve FAODV performansını değerlendirmek için kullanılanlarla aynıdır (bölüm 4.4.2). Bu bölümde, kullanılan senaryolar tanımlanır. .
Simülasyon bölgesi üzerine 50 devrenin dağıtıldığı deneyler için üç farklı senaryo düşünülmüştür. Tanımlanan senaryolar devre yoğunluğu ve bağlantı değişikliği açısından farklılık gösterirler. Bunlar aşağıdaki bölümlerde açıklanmıştır –

3.10.1. Darboğaz Senaryosu

Referans Noktası Grup Hareketlilik (RPGM) örneği darboğaz senaryosunu örneklendirmek için kullanılmıştır. Bu hareketlilik örneği içindeki parametreleri tablo 3.7'de gösterildiği şekilde tanımlarız 3.7 –

Tablo 3.7: Darboğaz senaryosu için Parametreler.

Parametreler	Değerler
Hareketlilik örneği	RPGM
Devrelerin Dağıtımı	Her grup içinde 10 5 grup
Simülasyon Bölgesi	2000 * 2000 m
Grup değişikliği olasılığı	0.25
Devre hızı	Maksimum hız: 5 m/s Minimum hız : 1 m/s
Grup merkezine maksimum uzaklık	50 m

Bu senaryo için nispeten seyrek yerleştirilmiş devre dizilerini düşünürüz. Her devre grup liderinden maksimum 50 metrede uzaklıkta dururken, devrelerin toplam sayısı 50'dir. Grup içinde değişiklik olduğu ile ilgili 0.25'lik bir olasılık bulunmaktadır. Örneğin, bu durum bir askerin ölümü ya da yaralı diğer askerlere yardım etmek için geçici taşınmadan olayı olabilir. Devrelerin maksimum hızı 5 m/s (tank gibi askeri araçları tanımlayabilecek) ve minimum hızı 1 m/s (askerlerin hareketi) olarak alınmıştır.

3.10.2. Kurtarma Operasyonu Senaryosu

Bu senaryo için bile, RPGM hareketlilik örneğini kullanırız. Bu senaryo nispeten küçük bir alanda çalışan işçi gruplarını temsil eder. Örneğin, çığdan kurtarma operasyonunda, küçük alan içinde iletişim kuran devreler kurabiliriz. Darboğaz senaryosundan çok nispeten daha yoğun devre dizisi düşünürüz. Devrelerin, darboğaz senaryosu ile karşılaştırıldığında grubu değiştirmek için (0.05) daha az bir olasılıkları bulunmaktadır. Bu senaryo için tanımlanan parametreler tablo 3.8'te gösterilmiştir.

Tablo 3.8: Kurtarma operasyonu senaryosu için parametreler.

Parametreler	Değerler
Hareketlilik örneği	RPGM
Devrelerin Dağıtımı	Her grup içinde 5 10 grup
Simülasyon Bölgesi	1000 * 1000 m
Grup değişikliği olasılığı	0.05
Grup merkezine maksimum uzaklık	100 m
Devre Hızı	Maksimum hız: 2 m/s Minimum hız : 1 m/s

3.10.3. Olay Menzili Senaryosu

Gauss Markov hareketlilik modeli [32] olay menzili senaryosunu örneklemek için kullanılmıştır. Bu örnekte, ayarlama parametresini değiştirerek rastgelelik derecesini değiştiririz. Deneylerimiz için, bu model içindeki hareketlilik derecesini tanımlamak için hız/açık güncellik frekansını değiştiririz. Parametreler tablo 3.9'te gösterilmiştir.

Tablo 3.9: Olay menzili senaryosu için parametreler.

Parametreler	**Değerler**
Hareketlilik Modeli	Gauss Markov Modeli
Devre sayısı	50
Simülasyon Bölgesi	500 * 500 m
Devrelerin maksimum gücü	5 m/s
Açı SD	0.5
Hız SD	0.5

Daha küçük bir simülasyon alanında bu senaryo için daha yüksek devre yoğunluğu düşünülmüştür. Örneğin, olayı geniş şekilde yazan basın habercileri arasındaki iletişimi tanımlayacaktır. Kurtarma operasyonu senaryosu ile karşılaştırıldığında devrelerin hareketliliği de daha yüksektir (5m/s). Açı ve hız standart sapma her biri için 0.5 olarak seçilmiştir.

4. BULGULAR

Tasarsızlerde güvenlik, darboğaz, olay menzili vs. gibi bazı yerleştirme senaryolarında çok önemlidir. Tasarsızler için olan geleneksel güvenliksiz yönlendirme protokolleri DoS, sahte e-posta, ön bellek zehirlenmesi gibi saldırıları önlemede başarısız olmuştur. Güvenli yönlendirme protokolleri tasarlamanın en önemli hedeflerinden biri, ağdaki tehlikeli devrelerin yol keşfi ve koruma mekanizmalarını bozmasını engellemektir. Ancak, bu durum eklenen güvenlik performans maliyeti ile birlikte gelir. Bu bölümde, FAODV oluşturan için performansı biz dizi senaryoya dayalı deney kullanarak AODV, SAODV, SEAD, DSDV ve DSR protokolleri ile karşılaştırılmıştır. Performans ve güvenlik arasındaki takas analiz edildi. Senaryolar, çelişen ihtiyaç eğiliminde bulunan darboğaz ve kurtarma operasyonları gibi tanımlı kritik gerçek dünya uygulamaları kullanırlar. Performans değerlendirme incelenmekte olan üç protokolün uygulanabilirliğine ilişkin bilgi verir ve hangi protokolün verilen senaryo için daha uygun olduğunu belirler.

4.1. GİRİŞ

Geçici ağlardaki güvenli yönlendirme literatürde kapsamlı şekilde incelenmiştir [13] [15] [16] [17] [19]. Geçici ağlar için güvenli yönlendirme protokolleri tasarlamak bir kaç sebepten dolayı zordur. Bir taraftan, protokol ağ bozulmasına karşı çoklu düzenlenmiş saldırılardan korumak zorundadır. Geçici ağlar, genellikle tüm devrelerin yönlendirme mekanizmasına katıldıkları açık bir çevre içine yayılmışlardır, tasarımcılar hem *aktif* hem de *pasif* saldırılara karşı önlem alma gibi sorunlarla karşılaşmışlardır [15]. Diğer taraftan, devreler genellikle, katı güç kuralları olan düşman ve barınması zor olan bölgelere yayıldığı için, yönlendirme protokolü güç bilinçli olmalıdır.

Bunun anlamı, güvenlik önlemleri uygulamak için kullanılan kriptografik temellerin hızlı ve etkili olması gerektiğidir. Güvenlik ve performans arasındaki takas, yönlendirme protokolünün tüm şartlar altında en uygun biçimde çalışması için analiz edilmelidir.

Tasarsızler için güvenli yönlendirme protokolleri performansının önceden analiz edilmiş olmasına rağmen, adı geçen araştırma için motivasyonda, daha yüksek durma sürelerinde birleşemeyen Rastgele Nokta hareketliliği örneği olarak düşünmüşlerdi [33]. Ayrıca, Rastgele Nokta hareketliliği, Tasarsız yayılımına ait bazı gerçekçi senaryoları yakalamada yeterli değil. Darboğaz, kurtarma operasyonu vs. gibi gerçekçi bir bölge içinde devrelerin hareketliliği örneklendirmek için, bazıları bölüm 4'te tanıtılmış olan daha çok yönlü hareketlilik örneklerine ihtiyacımız var. Bu bölümde, senaryoya dayalı deneylerimizin tasarımına odaklanacağız ve SEAD performansını analiz edeceğiz (bölüm 3). Performansını DSR , DSDV ,AODV SAODVve FAODV ile karşılaştıracağız. Özel yayılım senaryoları için performans ve güvenlik arasındaki takası analiz edeceğiz ve uyglanacak için .

Bölümün geri kalanı aşağıdaki şekilde düzenlenmiştir. – Bölüm 4.2'te, deneysel kurulumumuzu açıklıyoruz. Bölüm 4.3 karşı karşıya kalınan bazı sorunları ve bu sorunların nasıl çözüldüğünü tartışır. Kullanılan senaryo ve ölçümler Bölüm 4.4'te açıklanmıştır. Bölüm 4.6'de, elde edilen sonuçlar analiz edilir ve Bölüm 4'te gelecekteki çalışmalarımızdan bahsederek bu bölümü sonlandırırız.

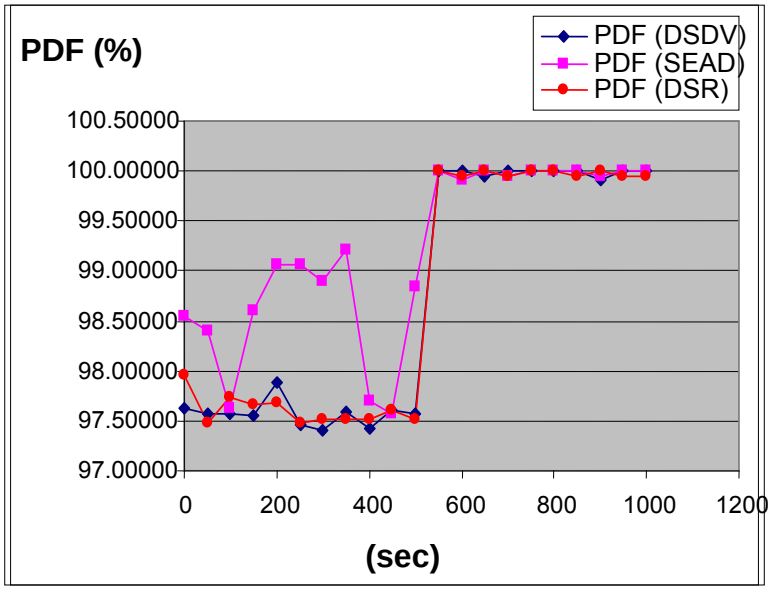
4.2. SONUÇLAR

Darboğaz ve yardım operasyonu senaryoları için durma sürelerini 0-1000 saniye olarak değiştirdik. Olay menzili senaryosu için, hareketlilik ölçümü olan hız ya da açı güncelleme frekansı olarak anılan Gauss Markov hareketlilik örneğinin parametresini değiştirdik. Her 5 saniyeden her 60 saniyeye kadar olan güncelleme frekansını değiştirdik. Üç ölüm için olan her senaryo etkisi seçilen üç protokolde araştırılmıştır.

4.2.1. Paket Teslim Bölümüne Etkisi (PDF)

Darboğaz senaryosu için, şekil 4.1'de gösterildiği gibi 100-400 saniye durma süresi için paket teslimatı açısından SEAD hem DSDV hem de DSR protokollerinden üstün gelir. Bu durum, DSDV tarafından SEAD'in kaçındığı yönlendirme tablosu güncelleme sayısını azaltmak için *ağırlıklı yerleşim gecikmesi* kullanıldığı gerçeğine dayanır. Dolayısıyla, SEAD genellikle verilen süre içinde DSDV'den daha yeni yollara sahip olur ve bu sebeple devrelerim, daha başarılı teslim paketleri anlamına gelir şekilde daha fazla güncel yönlendirme tabloları bulunmaktadır. Daha yüksek durma süreleri için (500

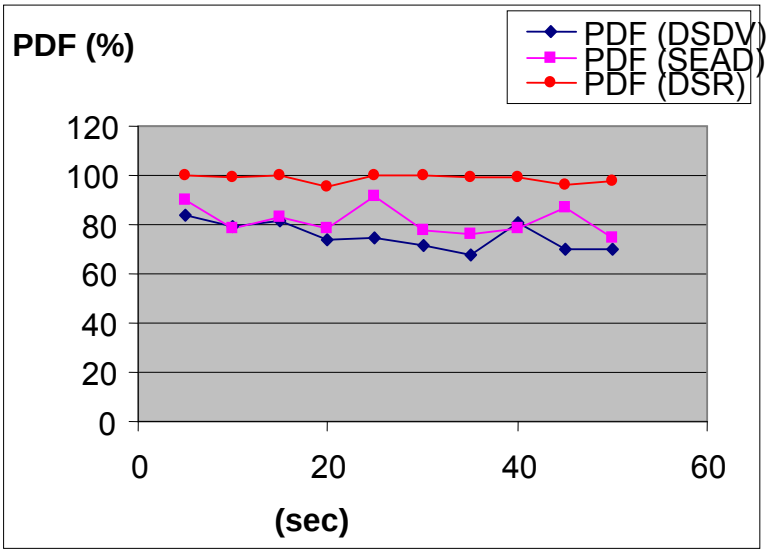
saniyeden fazla), devreler hemen hemen sabit olduğu için tüm üç protokol PDF'e neredeyse %100 sağlama yapmak için birleşirler ve bu sebeple ağ içindeki yoğunluk azalır.



`

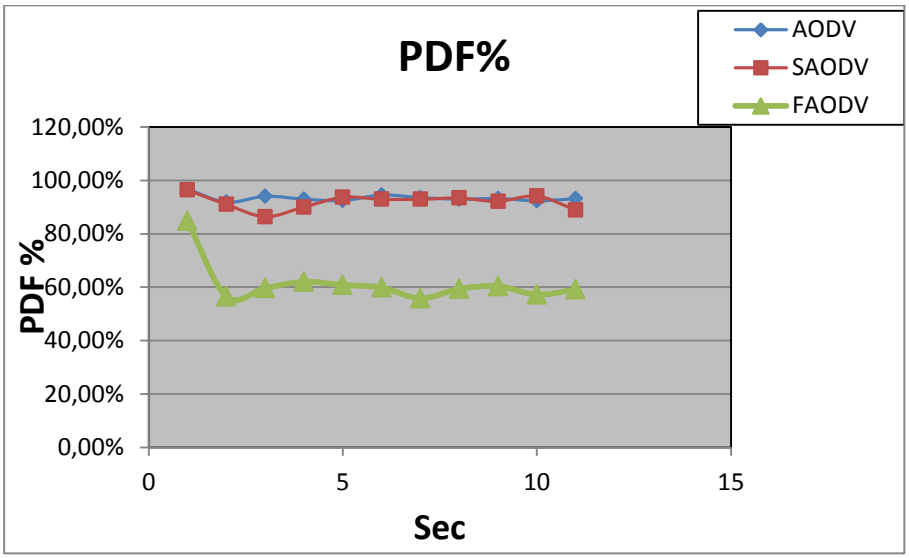
(Darboğaz: Durma süresi karşısında PDF) (Durma Süresi (saniye))

Şekil 4.1: Durma Süresi karşısında Darboğaz senaryosu için PDF.



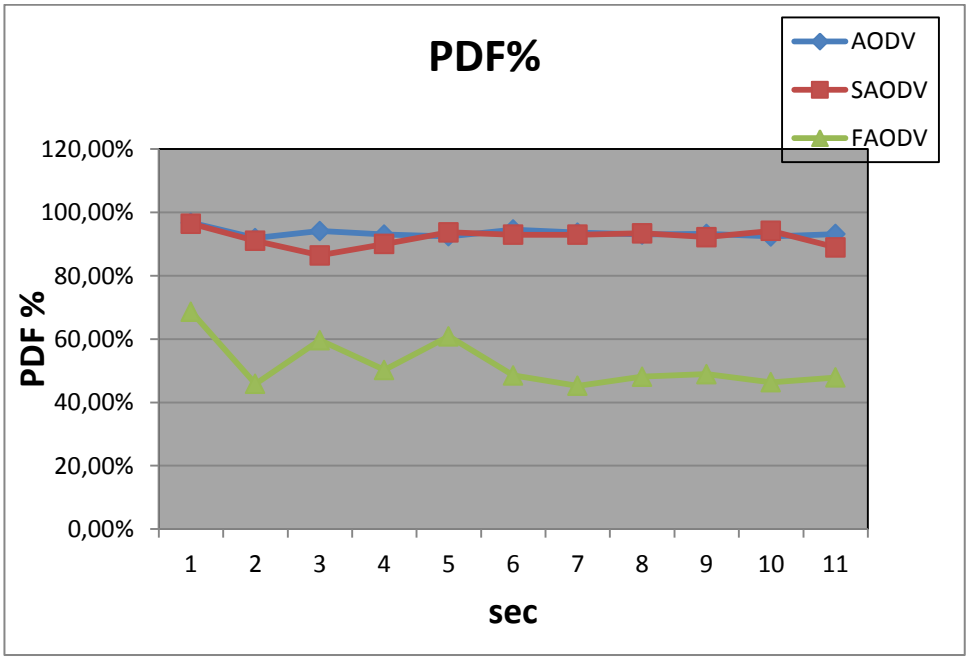
(Olay Menzili: S/A Güncelleme frekansı karşısında PDF) (Güncelleme Frekansı (saniye))

Şekil 4.2: Güncelleme Frekansı karşısında olay menzili senaryosu için PDF.



(Olay Menzili: S/A Güncelleme frekansı karşısında PDF) (Güncelleme Frekansı (saniye))

Şekil 4.3: Güncelleme Frekansı karşısında olay menzili senaryosu için PDF AODV FAODV ve SAODV.



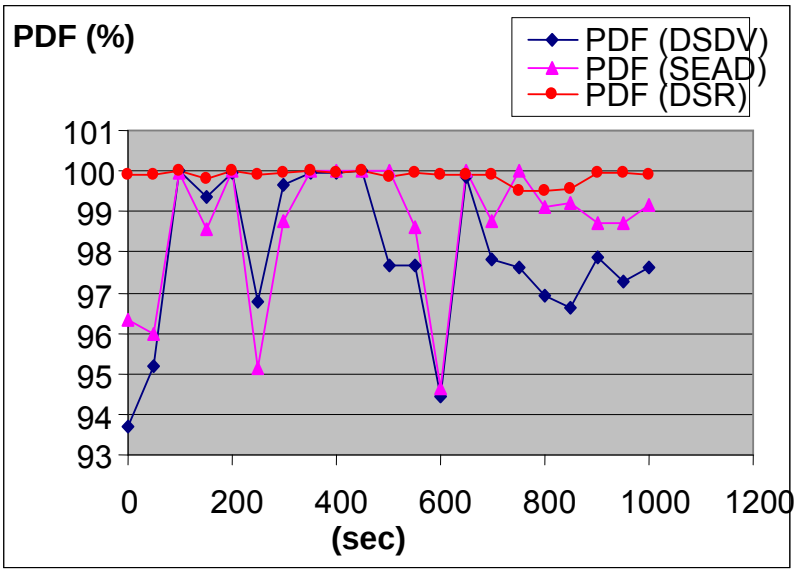
Şekil 4.4 : Güncelleme Frekansı karşısında olay menzili senaryosu için PDF AODV GÜVENLİĞİ FAODV ve SAODV.

Olay menzili senaryosu için, hız/açı güncelleme frekansı etkisi şekil 4.3'te gösterilmiştir. DSR protokolünün, SEAD ve DSDV ile karşılaştırıldığında çok yüksek PDF'si olduğu görülmüştür. Bunun sebebi DSR'nin reaktif protokol olmasıdır ve bu sebeple ağ içindeki değişikliklere proaktif protokol olan SEAD ya da DSDV'den daha iyi bir şekilde uyum sağlar. Olay menzili senaryosu, SEAD'in ağ içindeki bağlantı

değişikliklerine ve hareketliliğe DSDV'den daha iyi uyum sağladığını gösteren darboğaz senaryoları ile karşılaştırıldığında ağı daha yoğun devre dağılımı ve daha yüksek hareketlilik ile tanımlar.

Şekil 4.3'te gösterilmiştir .AODV ve SAODV ile karşılaştırıldığında çok yüksek PDF'si olduğu görülmüştür. FADOV 15 % daha düşük programlama metodu.

Şekil 4.4'te gösterildiği şekilde kurtarma operasyon senaryolarını düşündüğümüzde, DSR'nin tekrar hem SEAD hem de DSDV'den üstün geldiği ve hemen hemen %100 daha yüksek durma süresi ile bir PDF verdiği görülür. Buna karşılık, SEAD ve DSDV daha yüksek durma süresi için DSDV'den üstün gelen SEAD ile birlikte değişen performans gösterir. (700'den büyük).



(Kurtarma Operasyonu: Durdurma Süresi karşısında PDF) (Durdurma Süresi (saniye))

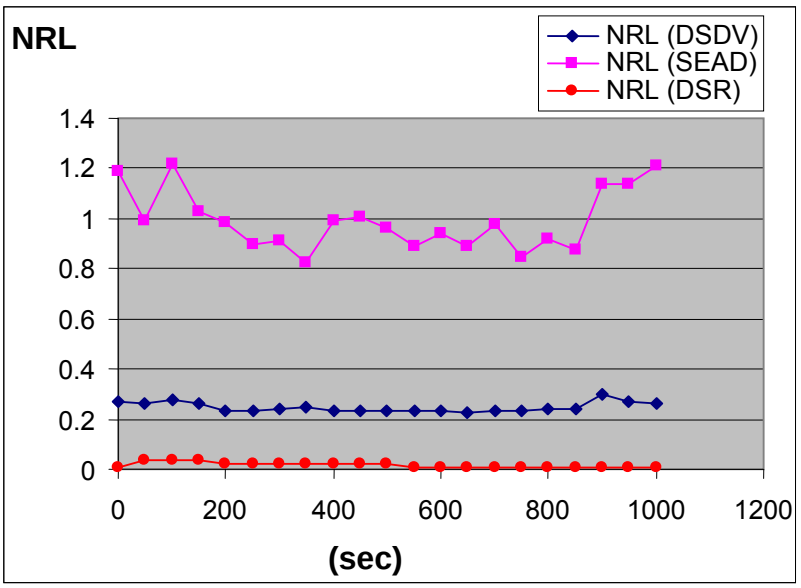
Şekil 4.5 : Kurtarma operasyonu için Durdurma Süresi karşısında PDF.

4.2.2. Standartlaştırılmış Yönlendirme Yükü üzerine Etki (NRL)

Şekil 4.6, 4.7 ,4.8 ve 4.9, üç senaryo için Standartlaştırılmış Yönlendirme Yükü ile ilgili değişim hareketliliğinin etkisini gösterir. Tüm senaryolar için, SEAD, DSR ve DSDV'den daha yüksek bir yönlendirme sabiti gösterir. DSR, reaktif bir protokol olmasından olayı en az üç sabit sahibidir, bu sebeple talep edildiğinde DSDV ve SEAD'de bulunan periyodik yönlendirme güncellerine karşılık sadece yolları tanıtır.

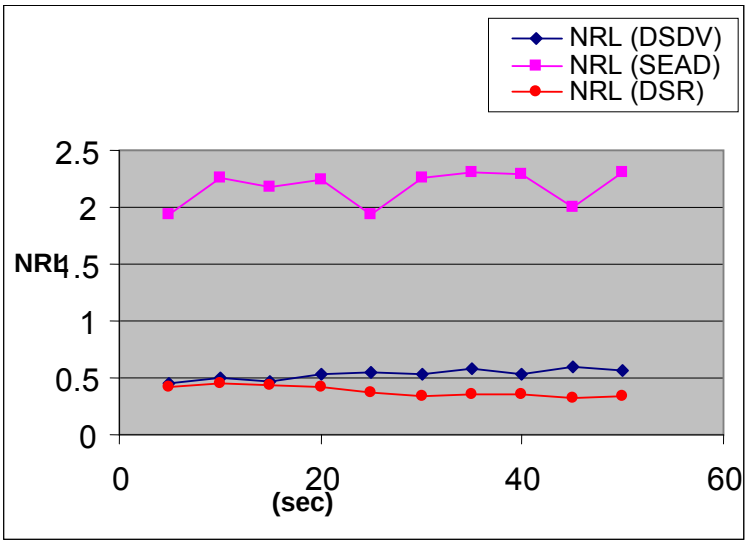
Devrelerin yoğunluğunun ağ içinde arttığı ve Standartlaştırılmış Yönlendirme Yükünün DSDV için SEAD arttığı görülmüştür. Bu durum, şekil 2.a, b ve c'den anlaşılabilir. olay

menzili senaryosu için olan yönlendirme yükü (devrelerin yüksek yoğunluğu) şekil 2.b'de gösterildiği şekilde 2 ve 2.5 arasında değişirken, darboğaz senaryosu şekil 2.a'da gösterildiği şekilde 0.8 ve 1.2 arasında değişir. Ancak, reaktif yönlendirme protokolünün devre hareketliliği açısından proaktif yönlendirme protokolünden daha uyumlu olduğunu tekrar vurgulaması ile, DSR üç senaryo boyunca yönlendirme yüklerinin sabit değerlerini gösterir.



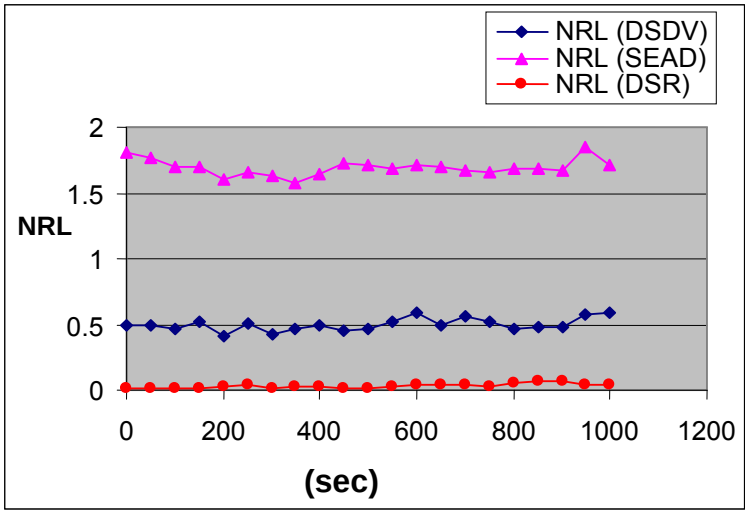
(Darboğaz: Durma süresi karşısında NRL) (Durma Süresi (saniye))

Şekil 4.6 : Durma Süresi karşısında Darboğaz senaryosu için NRL.



(Olay Menzili: S/A Güncelleme frekansı karşısında NRL) (Güncelleme Frekansı (saniye))

Şekil 4.7 : Güncelleme Frekansı karşısında olay menzili senaryosu için NRL.

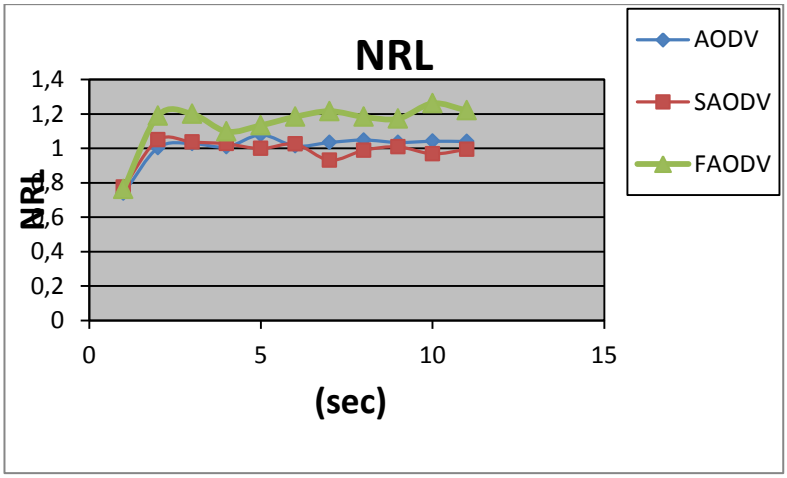


(Kurtarma Operasyonu): Durdurma Süresi karşısında NRL) (Durdurma Süresi (saniye))

Şekil 4.8: Kurtarma operasyonu için Durdurma Süresi karşısında NRL.

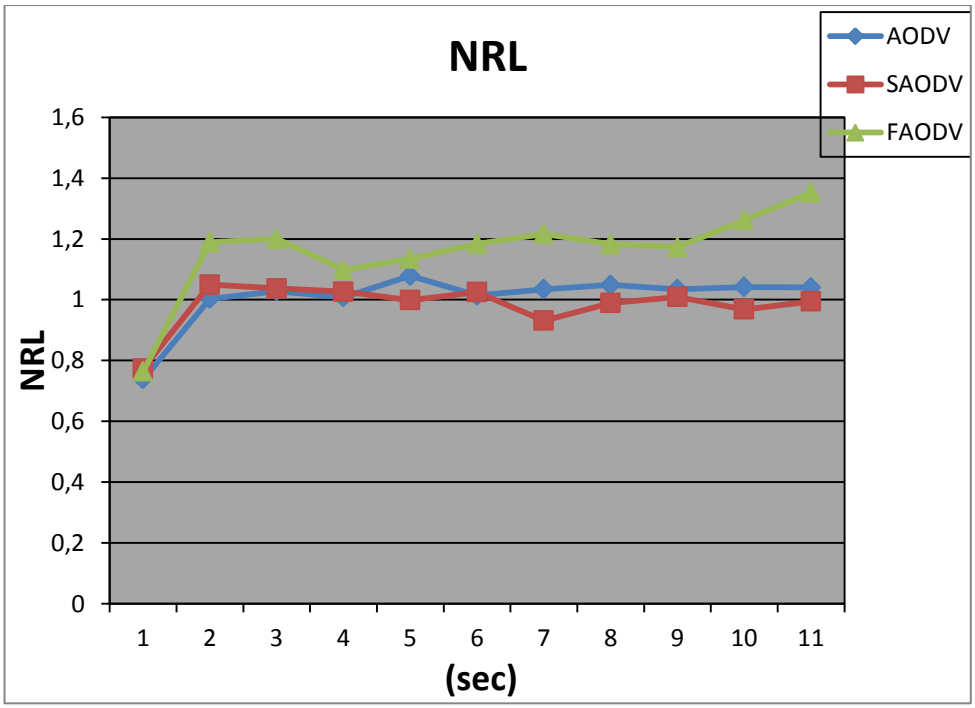
SEAD yönlendirme yükü, yerleşim gecikmesi yokluğunda devreler tarafından gönderilen yüksek sayıda yönlendirme tanıtımlarından dolayı tüm üç senaryo boyunca DSDV ve DSR'den daha yüksektir.

Şekil 4.9'te operasyonu için Durdurma Süresi karşısında NRL FAODV gösterildiği şekilde 0.8 ve 1.2 arasında değişir ve AODV ve SAODV gösterildiği şekilde 0.8 ve 1. arasında değişir cok fark arasındaki değil.



(Kurtarma Operasyonu): Durdurma Süresi karşısında NRL) (Durdurma Süresi (saniye))

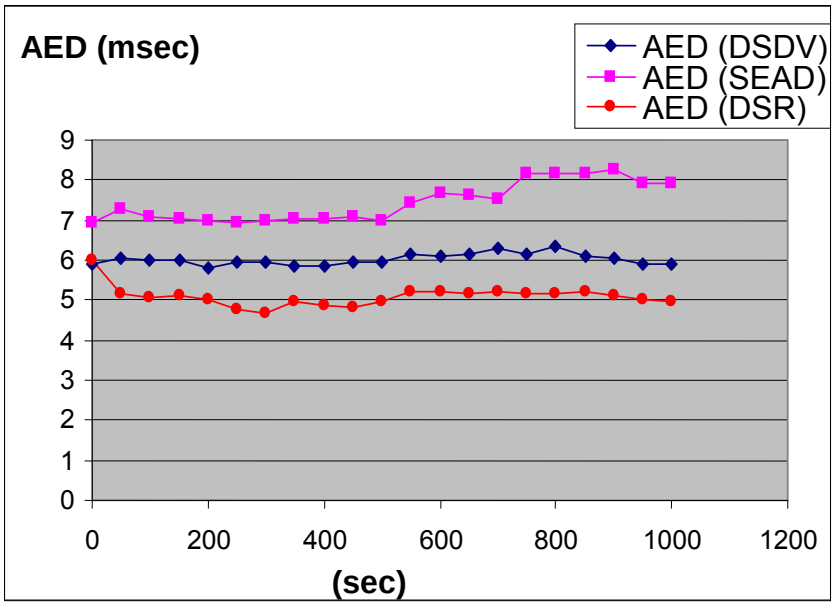
Şekil 4.9 : Kurtarma operasyonu için Durdurma Süresi karşısında NRL.



Şekil 4.10: Kurtarma operasyonu için Durdurma Süresi karşısında NRL wiith GÜVENLİĞ FAODV.

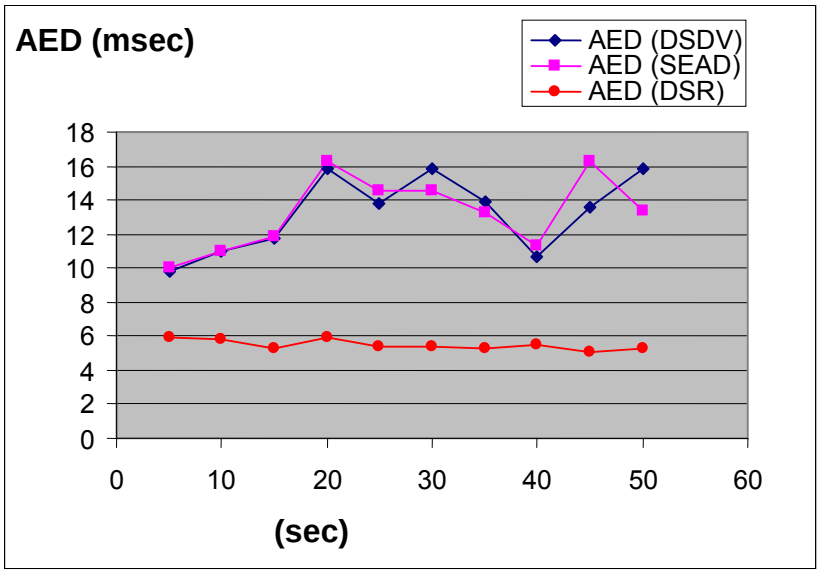
4.2.3. Ortalama Uçtan Uca Gecikme Etkisi (AED)

Şimdi ortalama uçtan uca gecikme olan en önemli ölçüm etkisini araştıralım. 4.11, 4.12 ve 4.13'da gösterildiği gibi, SEAD, DSDV ve DSR'den daha yüksek bir gecikme gösterir. Bu durum anlaşılabilir, çünkü yol doğrulaması için karışıklık hesabı her devrede işlem sabitine eklenir. Ayrıca, hareketlilik arttıkça ortalama uçtan uca gecikmenin de arttığını gördük. Darboğaz gibi düşük yoğunlukta bir senaryo için, 10 - 16 msec arasında değiştiği olay mahalli gibi daha yüksek yoğunlukta bir senaryo ile karşılaştırıldığında (şekil.4.9), gecikmenin 7-8 msec arasında değişerek SEAD için daha düşük olduğunu görürüz (şekil4.9).



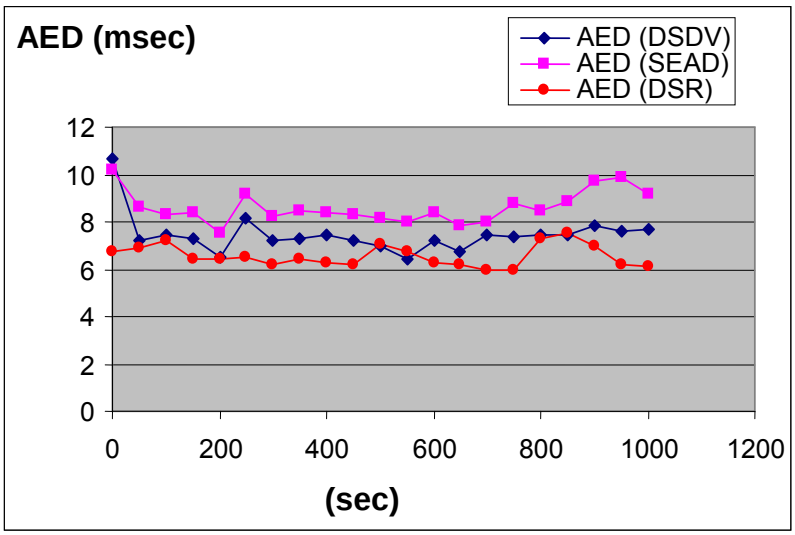
(Darboğaz: Durma süresi karşısında AED) (Durma Süresi (saniye))

Şekil 4.11: Durma Süresi karşısında Darboğaz senaryosu için AED.



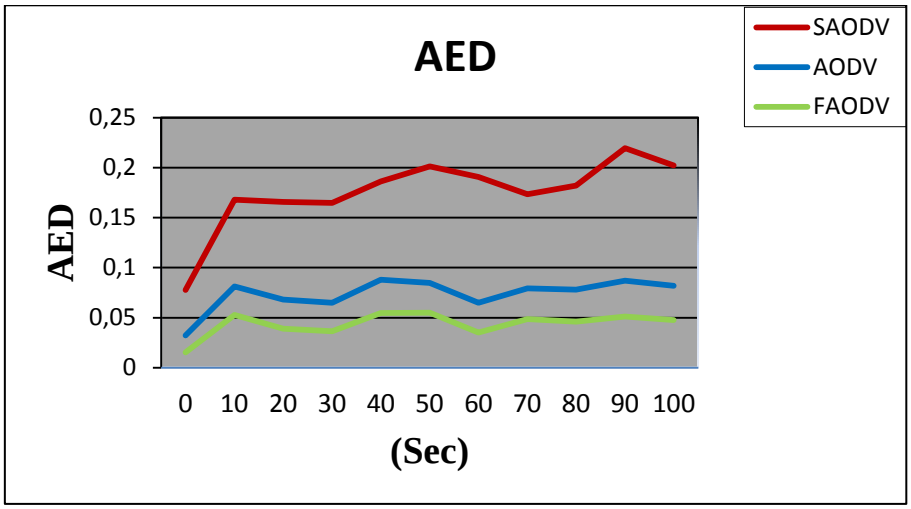
(Olay Menzili: S/A Güncelleme frekansı

Şekil 4.12: Güncelleme Frekansı karşısında olay menzili senaryosu için AED.



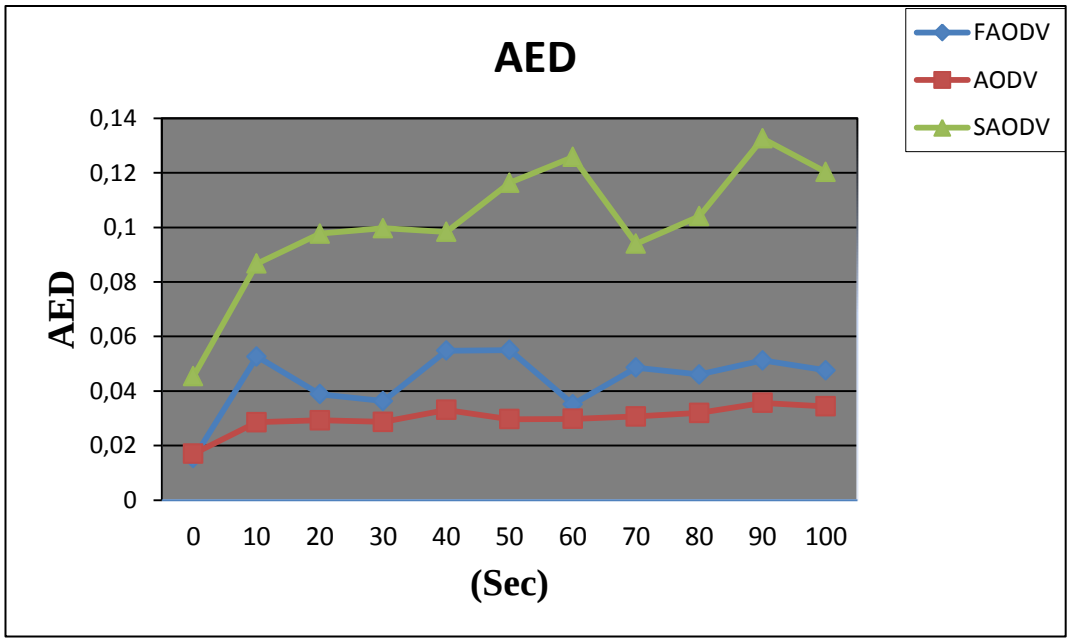
(Kurtarma Operasyonu: Durdurma Süresi karşısında AED) (Durdurma Süresi (saniye))

Şekil 4.13: Kurtarma operasyonu için Durdurma Süresi karşısında AED.



(Kurtarma Operasyonu: Durdurma Süresi karşısında AED) (Durdurma Süresi (saniye))

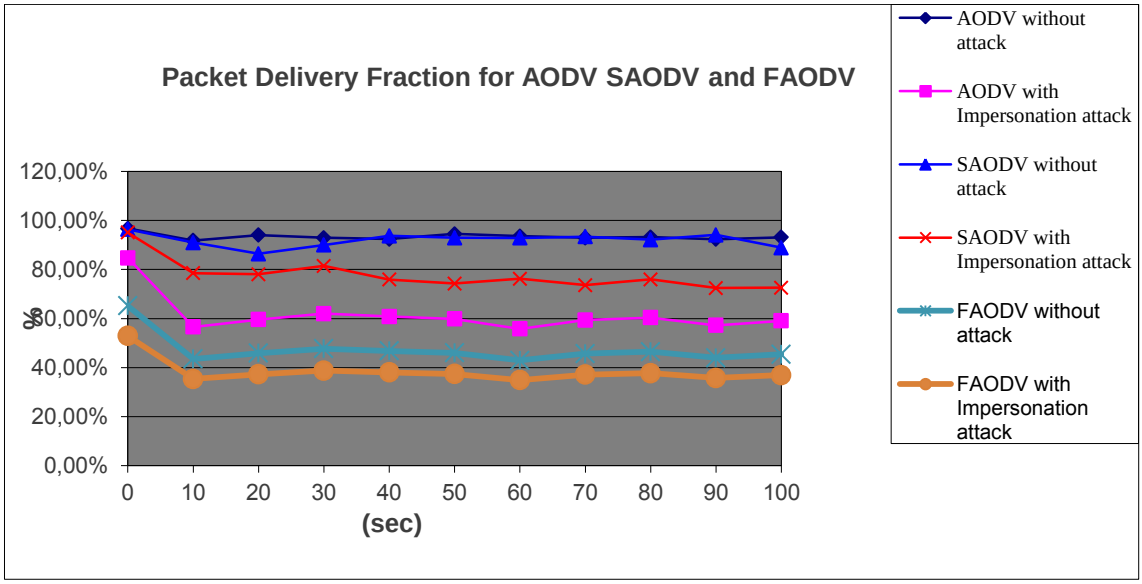
Şekil 4.14: Kurtarma operasyonu için Durdurma Süresi karşısında AED.



Şekil 4.15 : Kurtarma operasyonu için Durdurma Süresi karşısında AED with GÜVENLİĞ FAODV.

Değişen yükler kapsamında reaktif protokolün proaktif protokolden daha hızlı olma eğiliminde olduğu gerçeğini destekleyen grafiklerden görüldüğü gibi tüm üç senaryo boyunca DSDV ve SEAD'den daha düşük bir gecikme gösterir. [10] Bu durum gecikme ile ilgili katı bir üst sınır gerektiren çoklu ortam gibi uygulamalarda önemli olabilir. Dolayısıyla, DSR güvenlik konu olmadığında adı geçen uygulamalar için ideal bir seçenek olabilir.

Şekil 4.16: Kurtarma operasyonu için Durdurma Süresi karşısında AED. Değişen yükler kapsamında FAODV daha hızlı olma eğiliminde olduğu gerçeğini destekleyen grafiklerden görüldüğü gibi tüm üç senaryo boyunca SAODV ve AODV 'den daha düşük bir gecikme gösterir



Şekil 4.16: AODV ,FAODV VE SAODV PDF mesaj saldırı ve saldırsız karşılaştırma.

Biz şekle baktığımızda saldırı durumunda performans ve bazı olup olmadığını bir fark olduğunu bulmak AODV 20 %olması durumunda yaklaşık olarak , SAODV 10% , FAODV 5% son olarak gösteriyor.

.

5. TARTIŞMA VE SONUÇ

Kısa bir sonuçtan sonra, çalışmalarımızın sınırlandırmalarını yayınladık. Aynı zamanda çalışmalarımızla ilgili belirli olası uzantılar da önerdik. Bunlar değinmediğimiz ve büyük ihtimalle gelecekte araştırılacak bazı güvenlik yönleridir.

5.1. SONUÇ

Tasarsız bir uygulamadaki ağ topolojisi dinamik ve öngörülemez olabilir. Kablolu ağlarda kullanılan geleneksel yönlendirme protokolleri, bu tip dinamik ağlarda bazı ortak varsayımlar geçerli olmadığı için, çoğu kablosuz ağlara doğrudan uygulanamaz. Örneğin, bir varsayıma göre bir düğüm aynı alt ağ içinde diğerleri tarafından gönderilen yayın mesajını alabilir. Ancak, bu durum kablosuz mobil ağ içindeki düğümler için doğru olmayabilir. Bu çeşit ağ içinde bulunan bant genişliği genellikle sınırlıdır. Dolayısıyla, bu ağ örneği yönlendirme protokolleri için mükemmel zorluklar uygulamaktadır. Birçok Tasarsız yönlendirme protokolü önerilmiş bulunmaktadır. Önceki çalışmalar yeni protokoller tasarlama, var olan protokolleri kıyaslama ya da standart Tasarsız yönlendirme protokolleri tasarlanmadan önce protokolleri geliştirmeye odaklanmışlardır. Bu konudaki bir çok araştırma, belirli trafik profilleri ile birlikte rastgele ağlardaki ilgili yönlendirme protokolleri benzeşim çalışmalarına dayanır. Ancak, farklı araştırmalardan gelen simülasyon sonuçları sabit değildir. Bunun nedeni Tasarsız yönlendirme protokol modelleri, ağ ve kullanıcı trafik profillerini içeren uygulama çevrelerindeki uygunluk eksikliğidir. Dolayısıyla, yayınlanmış sonuç ve neticeler genellenemez ve kişinin Tasarsız uygulamasına uygun yönlendirme protokolü seçmesi için zorluk oluşturur.

Tasarsız ağlar için önerilen yönlendirme protokolleri, dinamik olarak değişen ağ topolojileri gibi temel gereklilikleri iyi karşılıyor gibi gözükmektedir. Ancak, güvenlik sorunları öncelikli olarak ihmal edilmiş olarak bırakılmıştır. Tasarsız ağlar yönlendirme protokollerinin doğrulama, bütünlük ve gizlilik bakış açısından güven altına alınmış olması gerekmektedir. Bu gereklilikler en azından güçlü doğrulama, şifreleme mekanizmaları, dijital imza, hesaba dayalı adresleme kullanarak kısmi olarak

karşılanabilir. Ayrıca, koruma araçları, yönlendirme için kullanılan yaklaşıma dayalı olarak her protokol için optimize edilebilir. Bazı Tasarsız ağ yönlendirme protokolü geliştiricileri, gerekli güvenlik hedeflerini yerine getirmek için protokol dahilinde IPSEC uygulamasını önermişlerdir. Daha önceki bölümlerde de tartışıldığı gibi bu çeşit bir yaklaşım tekrar çalma vs. sorunlarından dolayı tamamen yeterli değildir. Ayrıca, seviye bağlantılı şifreleme ya da çift yönlü tüneller gibi geleneksel güvenlik mekanizmaları Tasarsız ağların dinamik ve tahmin edilmez özelliğinden dolayı yeterli değildir. Tehlikeli düğümler tarafından gerçekleştirilen karışık hata olasılığı ve diğer kötü niyetli faaliyet olasılığı, gerekli anahtar yönetimi gibi soyut seviye güvence modeli ve güvenlik mekanizması seviyesinde yayılmış bir yaklaşım gerektirir. Yönlendirme protokolleri için bir güvenlik örneği önerdik. Dış saldırıları önceden önlemler alarak engellemek için bir şema sunduk. Belirli saptanabilir iç saldırılar tespit edilmiş ve onlar için sorumlu düğümleri ayırma amaçlı yöntemler sunulmuştur. Bu amaç için gerekli anahtar yönetim aynı zamanda belirli destek işlemleri içermeden gerçekleştirilmiştir. Uygulama gösterimizin sonuçları, şemamızdan kaynaklanan destek işlem etkisinin sınırsal olduğunu ve ağ performansı üzerinde göz ardı edilebilir etkileri olduğunu göstermektedir.

5.2. SINIRLANDIRMALAR

Şemamız FAODV protokolü için tasarlanmıştır ve istenildiğinde yönlendirme protokolleri ile iyi çalışır. Şema tablo-sürümlü bir protokol için tanıtılmış ise, destek işlemler çok yüksek olur, dolayısıyla ağ performansını etkiler. Bunun nedeni, tablo-sürümlü protokollerde, topolojide herhangi bir değişiklik özel bir yol kullanılmış olmasa dahi yayılmıştır. Topoloji bilgisi taşıyan bu mesajlar, doğrulanmış olmalıdır ve çok yüksek destek işlem ile sonuçlanmıştır. *Merhaba mesajlarının* değiş tokuş edildiği yerde şema herhangi bir protokol için uygun değildir. Yol talepleri gibi yayın paketleri dış bir düğüm ile tekrarlanmış ise, saldırı sadece cevap alındıktan sonra tespit edilir. Bu durum ağın gereksiz paket yollaması olarak sonuçlanır ve dolayısıyla ağ kaynaklarını tüketir. İç saldırıları tespit etme yöntemimiz saptanabilir saldırıları tespit etmede etkilidir. Ayrıca, düğümlerin işbirliği yapmadığını kabul etmekteyiz. Gerçekte durum böyle olmayabilir. İç saldırıların tespiti için şema ile ilgili daha fazla deney gerçekleştirilebilir.

5.3. GELECEK ÇALIŞMASI

Bu tezde, yönlendirme protokolü ile ilgili güvenlik sorunlarına değindik. Odak noktası, talep üzerine olan protokoller, özellikle FAODV olmuştur. Tablo-sürümlü protokollere özgü sorunları çalışmak ve onlarla bütünleşirken en uygun şekilde çalışan şemalara bakmak ilgi çekici olacaktır. İç saldırıların çok sınırlı bir analizini gerçekleştirdik. Özellikle saptanabilir olmayan bazı saldırılar kullanılmadı. İzinsiz girenleri tespit etmek amacıyla trafik profillerini analiz eden izinsiz giriş tespit şemaları araştırmak için zorlayıcı farklı bir alan olacak. Düğüm risklerinin ayrı durumlarını göz önünde bulundurduk. Düğümlerin sistemi yıkmak için birlikte çalıştıkları durumu incelemek çok zorlayıcı olacak. Tehlikeli düğümlerin tespit edilmesi özellikle dinamik olarak değişen ağlarda çok zorlayıcı bir sorundur.

KAYNAKLAR

[1]. ZHOU L., ZYGMUNT HAAS J., 1999, Securing Ad Hoc Networks, *Institute of Electrical and Electronics Engineers, Networks Special Issue on Network Security*, 13,(6).

[2]. MISHRA A., NADKARNI K., PATCHA A., 2004, Intrusion detection in wireless ad hoc networks Wireless Communications, *Institute of Electrical and Electronics Engineers*, 11(1), 48 – 60.

[3].YONGGUANG ZHANG, WENKE LEE, YI-AN HUANG, 2003,Intrusion detection techniques for mobile wireless Networks, *ACM Wireless Networks*,9(5).

[4].SANZGIRI K., LAFLAMME D., DAHILL B., LEVINE B. N., SHIELDS C., BELDING-ROYER E. M., 2005, Authenticated routing for ad hoc Networks, *IEEE Journal on Selected Areas in Communications*, 23(3), 598- 610.

[5]. JIM BINKLEY, WILLIAM TROST, 2001, Authenticated ad hoc routing at the link layer for mobile systems, *ACM Wireless Networks*, 7(2).

[6]. GUERRERO ZAPATA M., ASOKAN N., 2002, Securing ad hoc routing protocols, *proceedings of the ACM workshop on Wireless security*.

[7]. GEHRMANN C., NIKANDER P., 2000, Securing ad hoc services, *a Jini view, First Annual Workshop on Mobile and Ad Hoc Networking and Computing.*

[8]. YIH-CHUN HU, 2004, ADRIAN PERRIG, A Survey of Secure Wireless Ad Hoc Routing, *IEEE Security and Privacy*, 2(3), 28 – 39.

[9]. YI-AN HUANG, WENKE LEE, 2003, Intrusion detection: A cooperative intrusion detection system for ad hoc Networks, *proceedings of the 1st ACM workshop on security of ad hoc and sensor networks.*

[10]. LAKSHMI VENKATRAMAN, DHARMA AGRAWAL P.,(WCNC 2000), A Novel Authentication scheme for Ad hoc Networks, *Wireless Communications and Networking Conference, Institute of Electrical and Electronics Engineers*, l3,1268-1273

[11]. P. PAPADIMITRATOS AND HAAS Z., 2002, Secure routing for mobile ad hoc networks *(SRP) SCS Communication Networks and Distributed Systems Modeling and Simulation Conference*,27-31 http://wnl.ece.cornell.edu/ Publications / cnds02.pdf [Ziyaret Tarihi : 04 Kasım 2015

[12]. PERKINS C. AND ROYER E., 1999, Ad Hoc On-Demand Distance Vector Routing, *Proceedings 2nd Institute of Electrical and Electronics Engineers Workshop on Mobile Computing Systems and Applications (WMCSA'99).*

[13]. ALVARO A. CÁRDENAS, SVETLANA RADOSAVAC, JOHN S. BARAS, 2004, Ad hoc networks: Detection and prevention of MAC layer misbehavior in ad hoc networks, *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor Networks.*

[14]. ASAD AMIR PIRZADA, CHRIS MCDONALD, 2004, Establishing trust in pure ad-hoc networks, *Proceedings of the 27th conference on Australasian computer science* 26.

[15]. JOHNSON D. B. AND MALTZ D. , 1996, Dynamic Source Routing in Ad Hoc Wireless Networks. *T. Imielinski and eds. H. Korth, editors, Mobile Computing. Kluwer Academic Publishers*.

[16]. XIANJUN GENG, YUN HUANG, ANDREW B. WHINSTON, 2002, Defending wireless infrastructure against the challenge of DDoS attacks, *Mobile Networks and Applications*, 7(3).

[17]. SRDJAN CAPKUN, JEAN-PIERRE HUBAUX, LEVENTE BUTTYÁN, 2003, Security & transport: Mobility helps security in ad hoc networks, *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*.

[18]. YIH-CHUN HU, ADRIAN PERRIG, DAVID B., JOHNSON,2003, Secure routing: Rushing attacks and defense in wireless ad hoc network routing protocols, *Proceedings of the 2003 ACM workshop on Wireless security*.

[19]. HAO YANG, XIAOQIAO MENG, SONGWU LU, 2002, Self-organized network-layer security in mobile ad hoc networks, *Proceedings of the ACM workshop on Wireless security.*

[20].CAPKUN S.,BUTTYAN L., HUBAUX J.P.,2003 , Self-organized public-key management for mobile ad hoc networks, *Mobile Computing, Institute of Electrical and Electronics Engineers Transactions* on 2(1), 52 – 64.

[21]. SOYTÜRK M., HARMANCI A.E., ÇAYIRCI E., 2000, Gezgin Ad Hoc Ağlar ve Yol Atama, *Bilişim Zirvesi '01*, İstanbul, 4-7.

[22]. HAIYUN LUO, JIEJUN KONG, ZERFOS P., SONGWU LU, LIXIA ZHANG, URSA,2004: ubiquitous and robust access control for mobile ad hoc networks, *IEEE/ACM Transactions on Networking*,12,(6),1049 – 1063.

[23]. PANAGIOTIS PAPADIMITRATOS, ZYGMUNT HAAS J., 2003, Secure routing: Secure data transmission in mobile ad hoc Networks, *Proceedings of the 2003 ACM workshop on Wireless security.*

[24]. HAO YANG, HAIYUN LUO, FAN YE, SONGWU LU, LIXIA ZHANG,2004 , Security in mobile ad hoc networks: challenges and solutions, *Institute of Electrical and Electronics Engineers Wireless Communications*,11(1),38 – 47

[25]. TONY MANTUANO, JONGHOON JEONG, Survey and Comparison of Two Ad Hoc Routing Protocols, http://nislab.bu.edu/sc546/ sc546Fall2002/ adhoc/ diff.html [ZiyaretTarihi:04 Kasım 2015]

[26]. HU, Y.-C.; PERRIG, A.; JOHNSON, D.B.; 30 March-3 April 2003,Packet leashes: a defense against wormhole attacks in wireless Networks, *Twenty-Second Annual Joint Conference of the Institute of Electrical and Electronics Engineers Computer and Communications Societies, INFOCOM 2003,* Volume 3,1976 – 1986.

[27]. DAVIDH.SU Mobile Ad Hoc Networks(MANETs),http://w3.antd.nist.gov/ wahn_mahn.shtml [Ziyaret Tarihi : 04 Kasım 2015]

[28]. STAJANO F., ANDERSON R.J.,1999, The resurrecting duckling: Security issues for ad-hoc wireless networks" *In 7th Security Protocols Workshop,* Cambridge, United Kingdom, Springer-Verlag, Berlin Germany,1796.

[29]. Infrared Data Association, http://www.irda.org/ [Ziyaret Tarihi : 04 Kasım 2015]

[30]. RAYA M., HUBAUX J.-P., AND AAD I., DOMINO , 2004, A system to detect greedy behavior in *Institute of Electrical and Electronics Engineers* 802.11 hotspots, *Proceedings of the Second International Conference on Mobile Systems, Applications and Services (MobiSys2004), Boston, Massachussets.*

[31]. C. R. PERKINS, P. BHAGWAT, 1994, Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers, *ACM SIGCOMM, 234–244.*

[32]. REFIK MOLVA AND PIETRO MICHIARDI, 2003, Security in Ad hoc Networks, *Proceedings PWC (Personal Wireless Communications).*

[33]. YIH-CHUN HU, ADRIAN PERRIG, DAVID B. JOHNSON., September 2002 , Ariadne: A secure On-Demand Routing Protocol for Ad hoc Networks , *Atlanta, Georgia, USA ,MobiCom, 23-28,.*

[34]. HU Y-C., JOHNSON D. B., PERRIG A., 2003, SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks, *Ad Hoc Networks,* 1(1), 175-192.

[35]. PAPADIMITRATOS P., HAAS Z. J., 2003, Secure Link State Routing for Mobile Ad Hoc Networks,*International Symposium on Applications and the Internet USA ,MobiCom 146-153.*

[36]. NIE J., WEN J. , LUOVD J., 2006, An adaptive fuzzy logic based secure routing protocol in mobile ad hoc networks, *Fuzzy Sets and Systems*, 157(12), 1704-1712.

[37]. SEUNG Y., PRASAD N., ROBIN K., 2001, "Security-aware ad hoc routing for wireless networks*"*, Proceedings of the 2nd *ACM international symposium on Mobile ad hoc networking and computing*, Long Beach, CA, USA,

[38].PERRIG A.,CANETTI R., TYGARD D., AND SONG D.,2002, The TESLA Broadcast Authentication Protocol, *Cryptobytes, (RSA Laboratorie),* 5(2),2-13. http://www.rsa.com/rsalabs/cryptobytes/cryptobytes_v5n2.pdf[Ziyaret Tarihi: 04 Kasım 2015]

[39].SIVA RAM MURTHY C.,MANOJ B.S.,2004, *Ad Hoc Wireless Networks : Architectures and Protocols*, Prentice Hall Publishers, 013147023X.

[40].KIMAYA SANZGIRI, BRIDGET DAHILL, BRIAN NEIL LEVINE, CLAY SHIELDS AND ELIZABETH BELDING ROYER M.. November 2002 ,"A Secure Routing Protocol for Ad Hoc Networks" (ARAN) *In International Conference on Network Protocols (ICNP), Paris, France*. http:// citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.69.8105[Ziyaret Tarihi : 04 Kasım 2015]

[41].STEFANO BASAGNI, MARCO CONTI, SILVIA GIORDANO, IVAN STOJMENOVIC, "*Mobile Ad Hoc Networking*",, Wiley-IEEE Press, *Chapter 12: Ad hoc networks Security* Pietro Michiardi, Refik Molva, 0471373133. http://www.inrialpes.fr/planete/splash/PDF/pwc-survey.pd[Ziyaret Tarihi : 04 Kasım 2015]

[42].HONGMEI DENG, WEI LI, AND DHARMA AGRAWAL P.,2002, "Routing Security in Wireless Ad Hoc Network," *Institute of Electrical and Electronics Engineers Communications Magzine,* 40(10).

[43]. Internet X.509 Public Key Infrastructure Certificate and CRL Profile - *RFC 2459.*

[44]. WEIHONG WANG, YING ZHU, BAOCHUN LI. , 2003,"Self-Managed Heterogeneous Certification in Mobile Ad Hoc Networks ", *in the Proceedings of Institute of Electrical and Electronics Engineers Vehicular Technology Conference (VTC 2003),* Orlando, Florida, 10,6-9.

[45]. KONG J., ZERFOS P., LUO H., LU S., AND ZHANG L., November 11-14 2001, "Providing robust and Ubiquitous Security support for Mobile Ad Hoc Networks ", *Proceedings of the 9th International conference on Network Protocols (ICNP),* Riverside, California, USA.

[46]. EDITH C. H. NGAI AND MICHAEL R. LYU. 3(23-24), 2004, "Trust- and Clustering-Based Authentication Services in Mobile Ad Hoc Networks", *24th International Conference on Distributed Computing Systems Workshops* - W4: MDC (ICDCSW'04), Hachioji, Tokyo, Japan.

[47].MATEI CIOBANU MOROGAN, SEAD MUFTIC. January 27 - 31, 2003, "Certificate Management in Ad Hoc Networks", *Symposium on Applications and the Internet Workshops* (SAINT'03 Workshops, pp. 337.

[48]. DIRK BALFANZ, D. K. SMETTERS, PAUL STEWART AND H. CHI WONG: ,2002, "Talking To Strangers: Authentication in Ad-Hoc Wireless Networks", *Symposium on Network and Distributed Systems Security* (NDSS'02), Xerox Palo Alto Research Center, Palo Alto, USA.

[49].P.ZIMMERMAN,1995, MIT Press, *The Official PGP Users guide*, 0262740176.

[50].SCHNEIR B.: 1996 , *Applied CryptographyMethods*. John Wiley and sons inc,

[51]. DIERKS T. , ALLEN C.,1999,*The TLS protocol version 1.0* , RFC 2246,

[51].BING WU, JIANMIN CHEN, JIE WU, MIHAELA CARDEI: A survey of attacks and counter measures in mobile ad hoc networks, *Department of Computer Science and Eng, Florida, Atlanta university*.http://student. fau.edu /jchen8 /web/papers/SurveyBookchapter.pdf [ZiyaretTarihi : 04 Kasım 2015]

[52]. KENT S. AND ATKINSON R. , Nov.1998 , Security architecture for the Internet Protocol. Internet Request for comment, *Internet Engineering Task Force*, RFC 2401.

[53]. COLİN COOPER, RALF KLASİNG, AND TOMASZ RADZİK,2006,Searching for Black-Hole Faults in a Network Using Multiple Agents , *Springer-Verlag Berlin Heidelberg A. Shvartsman (Ed.):* OPODIS, 4305,318–330,

[54]. KAMARULARIFIN ABD. JALIL, ZAID AHMAD, JAMALUL-LAIL AB MANAN, 2011, Mitigation of Black Hole Attacks for AODV Routing Protocol The Society of Digital Information and Wireless Communications, *International Journal on New Computer Architectures and Their Applications* (ISSN: 2220-9085) (IJNCAA) 1(2): 336-343

[55]. VINAY P.VIRADIA, VIDHYA P.PATEL,2012, Simulation of AODV under Black hole Attack in MANET / *International Journal of Engineering Research and Applications (IJERA)* ISSN: 2248-9622,2(6),180-184.

[56]. JIWEN CAI, PING YI, JIALIN CHEN, ZHIYANG WANG, NING LIU, 2010, An Adaptive Approach to Detecting Black and Gray Hole Attacks in Ad Hoc Network, *24th Institute of Electrical and Electronics Engineers International Conference on Advanced Information Networking and Applications*

[57]. SCHNEIER B., 1994, *Applied Cryptography*,John Wiley & Sons, New York, ISBN: 978-1-119-096726-6.

[58]. MADHURYA M., ANANDA KRISHNA B., SUBHASHINI T.,2014, Implementation of Enhanced Security Algorithms in Mobile Ad hoc Networks. *Computer Network and Information Security*, (2), 30-37.

[59]. YUH-REN TSAI SHIUH-JENG WANG,2004, Routing Security and Authentication Mechanism for Mobile Ad Hoc Networks, *Institute of Electrical and Electronics Engineers*, 4716-4720.

[60]. Q. HE, D. WU, AND P. KHOSLA, 2005, Sori: A secure and objective reputation based incentive scheme for ad-hoc networks, *Wireless Communications and Networking Conference, Institute of Electrical and Electronics Engineers*, (4), 825–830.

[61]. HE Q., WU D., AND KHOSLA P, 2004, A secure and objective reputation based incentive scheme for ad-hoc networks. Wireless Communications and Networking Conference, 21-25 March *2004. WCNC. Institute of Electrical and Electronics Engineers*. 2,825–830.

[61]. JUNHAI LUO, MINGYU FAN, AND DANXIA YE, 2009, Black Hole Attack Prevention Based on Authentication Mechanism, *Institute of Electrical and Electronics Engineers*, 173-177.

[62]. ZHAO MIN ZHOU JILIU1,2009, Cooperative Black Hole Attack Prevention for Mobile Ad Hoc Networks, *Institute of Electrical and Electronics Engineers*, 5, 26-30.

[63]. GUOJUN WANG QIONG WANG JIANNONG CAO MINIYI GUO, 2007, An Effective Trust Establishment Scheme for Authentication in Mobile Ad Hoc Networks, *Institute of Electrical and Electronics Engineers*, 2, 749-754.

EKLER

EK 1
Tutorial for Performance Analysis of Ad Hoc Routing Protocols Using Ns-2

Getting your hands wet with ns-2

The ns-3.35 is available as an all-in-one package that includes many modules. Two modules that we will discuss in this tutorial are
ns-2 simulator. TCL/OTcl interpreter.

1.1. Procedure for installation ns-2 over ubuntu 12.10

Installing ns-2 can be time-consuming for beginners, especially when building it in a ubuntu environment. The detailed instructions for downloading and building ns-2 on ubuntu can be found at Christian's web page

1.2. Using ns-2

The following are a few useful tips on using ns-2 -
ns-2 running directory: in console, under directory:
toshiba@ubuntu:~/ns-allinone-2.35/ns-2.35$
Sample script files for wireless network simulations can be found under the directory:
toshiba@ubuntu:~/ns-allinone-2.35/codes$
Setting your environment –
In order to run your scripts from *any* directory, the PATH environment variables must be set. In order to do this, type the following at the command prompt as it is-
export ns HOME=toshiba@ubuntu:~/ns-allinone-2.35/ns-2.35$
export PATH=$ns_HOME/tcl8.4.5/unix:$ns_HOME/tk8.4.5/unix:$ns_HOME/ bin:$PATH
export LD_LIBRARY_PATH=$ns_HOME/tcl8.4.5/unix:$ns_HOME/tk8.4.5/unix:\ $ns_HOME/otcl-1.8:$ns_HOME/lib:$LD_LIBRARY_PATH
export TCL_LIBRARY=$ns_HOME/tcl8.4.5/library
Data files for ns-2: The input to ns-2 is a Tcl script file. Each script file corresponds to one specific experiment scenario and has the extension .tcl
To edit your script file use any Windows editor such as *editplus*, *notepad*, etc.
To run the script using ns type the following at the command prompt –
% ns filename.tcl
Some small scripts can also be run in command line mode. For this, just type ns and enter your commands line by line.

Procedure for Running Scenario-based Experiments

The procedure for running the scenario-based experiments are shown as a flow diagram are elaborated in the following sections

ÖZGEÇMİŞ

Kişisel Bilgiler

Adı Soyadı	Mohammad ABOSAALEEK
Uyruğu	ÜRDÜN
Doğum tarihi, Yeri	1968 , Amman
Telefon	05318820069/ 00962796421722
E-mail	Moh1_hammad@yahoo.com

Eğitim

Derece	Kurum/Anabilim Dalı/Programı	Yılı
Doktora	İ.Ü. Fen Bilimleri Enstitüsü/Bilgisayar Mühendisliği	2014
Yüksek Lisans	İ.Ü. Fen Bilimleri Enstitüsü/ Bilgisayar Mühendisliği	2006
Lisans	philadilphia university/ Computer Science & Information Systems	2001
Lise	High Secondary School Scientific Stream , Tawjihi	1986