IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 20, 2020, accepted May 30, 2020, date of publication June 11, 2020, date of current version June 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3001688

A Novel SIP Based Distributed Reflection Denial-of-Service Attack and an Effective Defense Mechanism

I. MELIH TAS[1], BASAK GENCER UNSALVER[1], AND SELCUK BAKTIR[2], (Member, IEEE)
[1]Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Bahçeşehir University, Istanbul, Turkey
[2]College of Engineering and Technology, American University of the Middle East, Kuwait

Corresponding author: Selcuk Baktir (selcuk.baktir@aum.edu.kw)

ABSTRACT We introduce a novel SIP based attack, named as the SR-DRDoS attack, that exploits some less known SIP features by using the IP-spoofing technique, the reflection based attack logic and the DDoS attack logic. Furthermore, we develop a SIP-based DoS/DDoS attack simulator, named Mr. SIP, and use it to implement our SR-DRDoS attack. Our attack is shown to dramatically increase the CPU load of a SIP server from 0% up to 100% in only 4 minutes after the attack is initiated. Since our intelligent attack creates legitimate traffic on the SIP network by using reflection methods, it bypasses black-lists as well as IP, packet-count or session/transaction based rate limiting and automatic message generation detection systems which exist in state-of-the-art security perimeters such as firewalls, intrusion detection/prevention systems and anomaly detection systems. Moreover, we propose a novel defense mechanism that effectively mitigates our proposed DRDoS attack. Our defense mechanism is shown to successfully reduce the CPU load of a SIP server under attack from 71% down to 18% within 3 minutes after it is initiated.

INDEX TERMS VoIP, voice over IP, VoIP security, SIP, session initiation protocol, SIP, SIP security, DoS, DDoS, DRDoS, distributed reflection denial of service attack, reflection attack.

I. INTRODUCTION

The Voice Over IP (VoIP) protocol has become an important component of modern corporate communications and many enterprises completely depend on it for their voice and video communication. However, VoIP brings opportunities along with its security risks due to existing vulnerabilities in the Internet Protocol (IP) and their possible exploitation by hackers [1]. The Session Initiation Protocol (SIP) is a widely-used VoIP signaling protocol for the signaling and control of multimedia communication sessions. The most common applications of SIP are Internet telephony and video calls over IP networks [2]. SIP defines the messages that govern the installation, termination and other basic elements of calls that take place between endpoints. The fraud survey report of the Communication Fraud Control Association (CFCA) shows that the estimated global telecom revenues for 2017 are $2.30 trillion and the estimated global loss is $29.2 billion for the same year [3]. Losses due to the abuse of network, device or configuration weaknesses are reported to be $1.29 billion.

Denial-of-Service (DoS) attacks prevent resources, e.g. servers, in a network from being accessed by users either temporarily or indefinitely. A DoS attack is performed by a single computer, whereas a Distributed Denial-of-Service (DDoS) attack is performed by multiple computers. In this attack, a vast amount of generated network traffic exhausts the server and prevents legitimate users from accessing its services [4], [5]. Unlike the Transmission Control Protocol (TCP) based applications, the User Datagram Protocol (UDP) based applications are not as mature and have some vulnerabilities. In the Distributed Reflection Denial-of-Service (DRDoS) attack, the attacker spoofs the victim's IP address and, using UDP, it sends a request for information to the reflectors who are known to respond to that type of a request [6], [7]. The reflectors answer the request for information and send (reflect) their response to the victim's IP address. The vulnerabilities in the retransmission mechanism of SIP are exploited using the IP spoofing technique in an earlier study [1]. In this study, we exploit the reflection mechanism in SIP. We attack SIP by exploiting the IP spoofing

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Imran Tariq.

technique and the ability to reflect request and response messages using appropriate SIP headers such as ''Via'' and ''Record-Route''. The faster and more efficient UDP is preferred for SIP communication, which serves better than the TCP for both server load reduction and improved call quality. Our work focuses on the application of reflection attacks to UDP based SIP services that would result in DRDoS attacks.

While DRDoS attacks have been investigated theoretically in the literature, they have not been implemented on a real SIP network to see their negative effects on the operation of the network. Furthermore, there are several attack simulators publicly and commercially available [8]–[17], whereas we are not aware of any simulators focusing on replicating multiple attack scenarios to help service providers and/or home users to test their networks for vulnerabilities. With this work, we propose a novel SIP based DRDoS attack, named as SIP Request Based DRDoS (SR-DRDoS), and show its efficacy in a real VoIP network environment using our novel attack simulator tool named Mr. SIP. Furthermore, we propose a novel defense mechanism that effectively mitigates the proposed DRDoS attack.

Our Main Contributions:

- We propose a novel SIP-based DRDoS attack, named as SR-DRDoS, which uses attack vectors obtained by merging the weaknesses of some less known SIP features with the IP-spoofing technique, reflection based attack logic and DDoS attack logic. To the best of our knowledge, our SR-DRDoS attack is the first real life example of a SIP request based reflection attack in a SIP network.
- We develop a novel attack simulator, named Mr. SIP, and use it in our VoIP/SIP security laboratory to implement our SR-DRDoS attack. Our SR-DRDoS attack implementation is shown to dramatically increase the CPU load of a SIP server from 0% up to 100% within 4 minutes after the attack is initiated.
- Since our intelligent attack method creates legitimate traffic on the SIP network by using reflection methods, it proves to bypass black lists as well as IP-based, packet count or session/transaction based rate limiting systems, and automatic message generation detection systems in the existing state-of-the-art security perimeters such as firewalls, intrusion detection/prevention systems and anomaly detection systems.
- Against the SR-DRDoS attack, we propose an effective defense mechanism which periodically collects a window of network traffic and calculates dynamic threshold values to trigger rule-based filtering actions. We show that it successfully reduces the CPU load of a SIP server under attack from 71% down to 18% within 3 minutes after it is initiated.

II. BACKGROUND

SIP is an application layer protocol that is designed to be independent from the protocol used at the transport layer. It is text-based and encompasses many elements of the Hypertext Transfer Protocol (HTTP) and the Simple Mail Transfer Protocol (SMTP) [18]. SIP works together with several other application layer protocols that define and transport the session media. Media identification and negotiation is done by the Session Description Protocol (SDP). For transmission of media (audio or video) streams, SIP typically uses either the Real-time Transfer Protocol (RTP) or the Secure Real-time Transfer Protocol (SRTP). For secure transmission of signaling messages, it is suggested that SIP messages are encrypted using the Transport Layer Security (TLS) protocol. VoIP, which includes SIP as an underlying signaling protocol, is an IP based technology, therefore data transmission with VoIP is vulnerable to the entire set of threats that are applicable to an IP network. Hence, there is an inherent security risk for SIP systems.

DoS attacks are named as one of the most alarming threats for the Internet [6]. They can be directed towards any network element to disrupt system functionality or networking capabilities [19]. DoS attacks are typically performed on a victim server [20]. They can be applied to SIP systems in five categories as Legitimate Message Flooding, Invalid Message Flooding, DRDoS, Malformed Messages and Spoofed Messages attacks [21]. While DoS attacks are performed by a single computer, DDoS attacks are performed by multiple computers [22], [23]. In DoS/DDoS attacks, the network stream coming to the victim server forces it to shut down. While the individual Internet packages of the attack are not malicious, they can consume the victim's network resources. The end result is valid network traffic, not categorized as harmful raid, which exhausts the server and prevents real users from accessing its useful services [4].

Several SIP based DoS attacks and related defense mechanisms have been proposed in the literature. In [46] the authors suggest a Bayesian change point model in order to detect attacks using a real-time traffic simulator based on social network modeling. In [47], the authors are able to identify DDoS attacks and attackers at the same time with their proposed change-detection algorithm implemented on the server-side. During their attack simulation, the five basic types of SIP messages (INVITE, REGISTER, OPTIONS, CANCEL and BYE) are used with randomly selected users as attackers at various levels of intensity. In [48], the authors propose a multiattribute flooding attack method on ten attack occasions, where four basic types of SIP messages (INVITE, 200 OK, ACK and BYE) are used simultaneously. In a similar study [49], the authors analyze two types of attacks, general DDoS floods and tailored DDoS floods. While the general attacks are created with botnets without prior knowledge of the system, the tailored attacks are specifically designed to bypass the rate-limiting rules by estimating the pre-defined thresholds on firewalls.

Reflection attacks exploit the challenge-response authentication mechanism that use the same protocol in both directions. The main idea in reflection attacks is to trick the target to respond to the attacker's challenge. The DRDoS attack

is a type of DoS attack where multi-point communication is achieved between the source and destination. DRDoS attacks are similar to DDoS attacks, however they differ in that they use multiple attacking computers and multi-point communication takes place between the target computer and the attacking computers. As a result, a much larger size of communication takes place in a DRDoS attack, and hence, it is significantly more intense than other types of DDoS attacks and can easily damage data or cause a server to crash. Reflectors between the source and destination make DRDoS attacks significantly more aggressive than traditional DoS/DDoS attacks [24]–[26].

A. DoS AND DDoS ATTACKS AGAINST SIP

Some mechanisms in the SIP protocol structure potentially facilitate exploitation using DoS/DDoS attacks. DoS attacks are service-blocking attacks which can be employed to consume excessive amounts of resources such as the bandwidth, physical disk space and CPU time. This may result in the corruption of configuration information, overloading with service requests (which the server cannot handle) or even the failure of the physical components in the network [4].

The total number of requests sent from any single IP address within a certain pre-determined time frame can be constrained, or session/transaction based rate-limiting protection can be employed, to protect SIP systems against DoS attacks. In order to circumvent such protection mechanisms, an attacker can perform a DoS attack by sending an excessive number of requests from distinct sources simultaneously, i.e. by using falsified IP addresses which is called IP spoofing. In this case, the attack is called a SIP-based DDoS attack.

Both DoS and DDoS attacks are attempts to prevent machines or users from accessing network resources. DDoS floods the target system and results in a high network traffic by preoccupying resources with the use of multiple attacker systems.

B. REFLECTION BASED ATTACKS

In the computer security world, the reflection attack is known as an attack type which exploits a challenge-response authentication protocol when the protocol runs the same way and sends similar messages in both directions. Here, the goal of the attacker is to mislead the other parties to respond to their impersonated challenges. The general attack flow for the reflection attack is as follows:

1) The attacker initiates a connection to target A.
2) Target A sends a challenge to authenticate the attacker.
3) The attacker establishes another connection to target B and sends target A's own challenge.
4) Target B responds to target A's challenge.
5) The attacker sends this response to target A in response to target A's challenge.

If the authentication mechanism is not carefully designed, target A will accept the response as valid and thus the attacker will have access to a fully authenticated connection channel [27].

SIP signaling is susceptible to reflection based attacks due to its design nature and these attacks can be replicated at the SIP level by forcing the proxy to forward messages to victims. Also, due to the mobile nature of VoIP, SIP clients do not use static IP addresses which makes IP based protection mechanisms such as blacklist/whitelist protection ineffective.

C. IP-BASED DDoS REFLECTION/AMPLIFICATION ATTACKS

In a DDoS reflection/amplification attack, the attacker spoofs the victim's IP address and sends traffic to the broadcast IP address [28]. When routers receive these data packets that are destined for the broadcast IP address, they transmit them to all hosts in their networks instead of a specific address [29]. In this attack, the attacker uses the broadcast network and has the advantage of having the ability to use a zombie without having to infiltrate or manipulate systems. The attacker can manipulate other hosts, called agents, to better hide its trace [4], [29], [30].

D. DISTRIBUTED REFLECTION DENIAL OF SERVICE ATTACKS

Reflection based attack logic is used in DRDoS attacks. In the DRDoS attack, the attacker aims to force reflectors to send packets towards the victim. To this end, the attacker impersonates the target victim and sends forged requests to millions of computers, resulting in the target victim being flooded by the responses from those computers. In a DRDoS attack, the effect of a single package sent by an attacker is reflected by many reflectors which makes it more effective than general DoS/DDoS attacks. The larger the number of reflectors is, the greater the effect of the attack becomes. For example, when a single package is sent over 250 reflectors, it becomes 250 times more resource consuming than a DDoS attack. The DRDoS attack is significantly more intense than other types of DoS attacks and can easily make a server collapse. DRDoS attacks can be prevented by having a large number of dominant victim nodes available for service [21]. Existing DRDoS intrusion detection mechanisms are designed for specific protocols, such as DNS and SNMP, but they are not widely used for SIP [31]–[34].

III. OUR VoIP/SIP SECURITY LAB ENVIRONMENT AND ATTACK SIMULATOR MR.SIP

We implement our DRDoS attack in our laboratory environment for VoIP/SIP security which includes specific software and applications. In our VoIP/SIP security laboratory, which implies an enterprise grade unified communication environment, we are able to achieve the SIP registration, session initiation and termination processes. On an Oracle VirtualBox virtualization environment, we have installed the required operating systems. We use many tools and utilities that are included in Kali. We use Trixbox as the target SIP-PBX platform. For the Trixbox server, in a virtualization environment, we reserve a CPU core and a 512 MB memory on a Macbook laptop with 4 CPU cores on an Intel Core i5 processor running

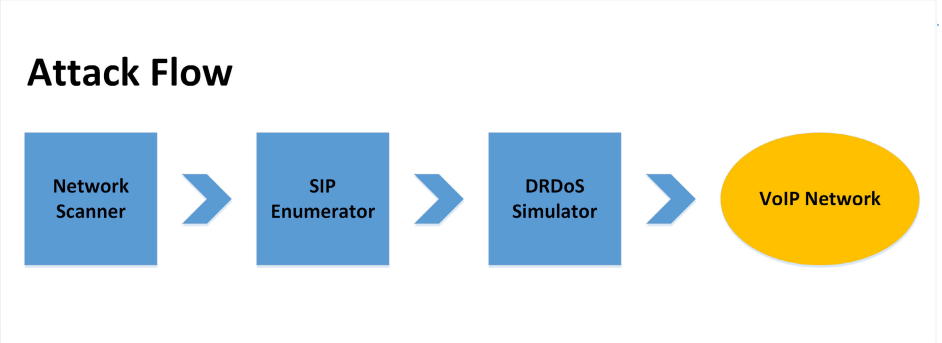

FIGURE 1. SR-DRDoS attack flow.

at 2.6 GHz and with an 8 GB DDR3 memory running at 1600 MHz clock speed. We use Zoiper [35] and X-Lite as our test clients registered to Trixbox SIP-PBX. We use ngrep for achieving the necessary quick network capture requirements [36].

A. OUR ATTACK SIMULATOR MR. SIP

We develop an attack tool, named Mr. SIP, and use it for our attack tests. We use Mr. SIP both as a SIP client simulator and a SIP traffic generator. Mr. SIP comprises four modules described as follows.

SIP-NES: This is the *network scanner* module of Mr. SIP. It takes IP range or IP subnet information as input and sends SIP OPTIONS messages to each IP address in the subnet. Based on its received responses, it outputs the list of potential SIP clients and servers in the specified subnet.

SIP-ENUM: This is the *enumerator* module of Mr. SIP. It sends REGISTER messages to each client IP address provided by the SIP-NES module. Based on the responses coming from the network, it outputs the list of valid SIP users in the subnet.

SIP-DAS: This is the *DoS attack simulator* module of Mr. SIP. It is developed to simulate SIP-based DoS attacks and comprises four components: spoofed IP address generator, SIP message generator, message sender and scenario player. It takes as input the outputs of the SIP-NES and SIP-ENUM modules along with some predefined files. SIP-DAS basically generates legitimate ''INVITE'' messages and sends them to a target SIP component via TCP or UDP. It has its own spoofed IP address generator in order to spoof IP addresses in the OSI application and network layers. It has three different options for spoofed IP address generation: manual, random or by selecting spoofed IP address from the subnet. IP addresses could be specified manually or they can be generated randomly. Furthermore, in order to bypass Unicast Reverse Path Forwarding (uRPF) filtering, which blocks IP addresses that do not belong to the subnet from passing into the Internet, we design a spoofed IP address generation sub-module. Our spoofed IP generation sub-module calculates the subnet used and randomly generates messages with spoofed IP addresses so that messages appear to come from within the subnet. In order to bypass automatic message generation detection (anomaly detection) systems, random ''INVITE'' messages are generated that contain no patterns. Each generated ''INVITE'' message is grammatically compatible with SIP RFCs and acceptable to all SIP components.

SIP-ASP: This is the *attack scenario player* module of Mr. SIP. It allows us to develop and implement various SIP-based DoS attack scenarios through the use of the SIP-DAS module as the framework. In this work, using SIP-ASP, we implement our ''SIP Request Based Distributed Reflection DoS'' attack scenario and try to deplete the resources of a SIP server and its clients.

B. SR-DRDoS ATTACK FLOW

DRDoS attacks require network scanning and enumeration capabilities for the preliminary steps of the attack [37]. Figure 1 shows the main components of our SR-DRDoS attack flow which are achieved through different modules of Mr.SIP, namely Network Scanner, SIP Enumerator and DRDoS Simulator. Our DRDoS attack simulator bypasses attack detection and prevention systems using its features such as IP spoofing, SIP request/reflection message generation and random SIP message generation.

A successful DoS attack against a VoIP network requires the IP addresses of the SIP server and the SIP clients, and the IDs of the SIP users [38]. We obtain these parameters using our attack tool Mr.SIP's Network Scanner and SIP Enumerator modules. The flowchart for the operation of our SR-DRDoS attack simulator is shown in Figure 2.

The Network Scanner module of Mr. SIP runs with the target IP address range as input and gives the SIP component IP address list as output. It sends a SIP ''OPTIONS'' message over the UDP port 5060 to all IP addresses within the given IP address range. By checking the response messages, it finds out whether there is a SIP component at the IP address. If the returned message is ''200 OK'', the device is potentially a SIP component. In this way, we can obtain the SIP component IP address list.

We provide the output of the Network Scanner module as an input to the SIP Enumerator module. In the next step, we obtain the SIP user information that is defined in the SIP

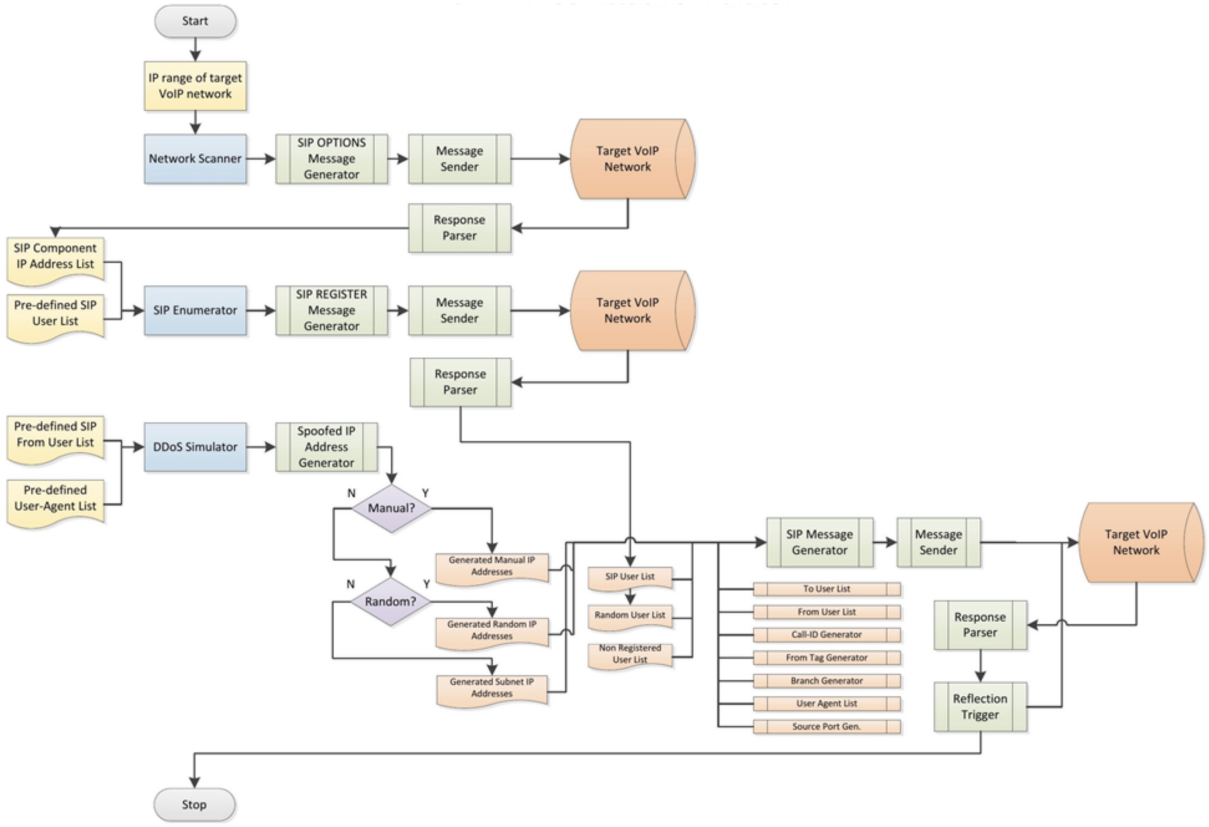

FIGURE 2. SR-DRDoS attack simulator.

system for each SIP component. For this process, the SIP Enumerator module sends SIP ''REGISTER'' messages to each SIP component for each user definition in the predefined SIP user list. If the returned message is ''401 Unauthorized'', this means there is such a user. If a ''404 Not Found'' message is returned, then there is no such user. In this way, we obtain the list of SIP users.

The output of the SIP Enumerator module is provided as an input to the DRDoS Simulator module. The SIP DRDoS Simulator module consists of four main parts: Spoofed IP Address Generator, SIP Message Generator, Message Sender and Reflection Trigger. It uses the ''Predefined from user list'' and the ''Predefined user agent list'' in addition to the ''SIP user list'' from the SIP Enumerator.

We use IP spoofing techniques to bypass the rate-limiting mechanisms in security defense systems such as firewalls and intrusion detection/prevention systems. We use spoofed IP addresses in the sent packets, so that packets are actually sent from a single source while appearing to come from multiple IP addresses at the network and application layers. IP spoofing is possible with the unreliable and connectionless UDP protocol. We design our attack simulator code to realize IP spoofing with three different options: counterfeit IP addresses can be generated manually, randomly or from the same network subnet that the attack was originated from. If uRPF filtering is available in the network where the attack is initiated, it will not allow us to have packets originating from an IP address outside the subnet. To avoid this situation, we have added an option in our attack simulator that allows us to generate random IP address from the same subnet.

In order to bypass anomaly detection systems that detect automatic message generation, our attack simulator generates SIP ''INVITE'' and ''REGISTER'' messages with random values. Since each generated SIP message is grammatically compliant with SIP RFCs, it is accepted by SIP components [39].

Our attack simulator's SIP ''INVITE'' message generation mechanism places the target user's information in the ''To'' header. The attack can be performed for a single user or for all legitimate users, and it can affect the whole SIP server system. The ''Via'', ''User-agent'', ''From'' and ''Contact'' headers in the SIP ''INVITE'' message are generated with the values from previously prepared and/or enumerated lists. The ''From Tag'', ''Branch'' and ''source-port'' fields in the ''Via'' and ''Call-ID'' parameters/headers are filled with rationally generated random values. In addition, the IP addresses in the ''Contact'' and ''Via'' headers are filled with fake IP addresses generated by the IP spoofing sub-module of our simulator. The generated packets are transmitted to the target system via the message sender sub-module using UDP. The responses are passed through the response parser sub-module to reflect the attack with the reflection trigger

sub-module. More information on the implementation of the attack is given in Section IV.

IV. SIP REQUEST BASED DISTRIBUTED REFLECTION DENIAL-OF-SERVICE ATTACK

The impact of DoS attacks on VoIP systems varies greatly according to the nature of the victim [40]. When the attack is targeted towards the user, the user's phone will be out of service. On the other hand, when the attack is targeted to a SIP proxy, all users that are making/receiving calls through the proxy will be out-of-service [21], [30].

SIP applications should be accessible over a wide area due to practical needs. However, SIP devices are vulnerable to spoofing and can be used as reflectors in DDoS attacks [41]. As described in Section II-C, reflection attacks can exploit this weakness by ensuring that SIP messages can be delivered to any target victim over any SIP proxy [21], [42]. In this paper, we propose and implement a ''SIP Request Based Distributed Reflection Denial-of-Service'' attack, named SR-DRDoS, which exploits this vulnerability.

A. "VIA" AND "RECORD-ROUTE" HEADERS

In our proposed ''SR-DRDoS'' attack, we exploit the ''Via'' and ''Record-Route'' headers in SIP. Here, we describe how these headers work in usual call scenarios. When a User Agent Client (UAC) creates a SIP request, it has to include the ''Via'' header which defines the protocol name (SIP), the protocol version (2.0), the transmission type (UDP or TCP), the UAC IP address and the protocol port number used in the request (typically 5060). If there is already a ''Via'' header in the message, the UAC adds the new entry to the top of the list before sending the message to the next hop, so that the User Agent Server (UAS) can respond to the correct device [43]. For example, if a SIP soft-phone sends an ''INVITE'' request, the ''Via'' header will appear as follows:

Via: SIP/2.0/UDP 10.11.228.67:5060

If an ''INVITE'' message is sent, it may contain more than one ''Via'' header when it reaches the called device. When the called party is ready to send the ''100 Trying'' message, it deletes the topmost ''Via'' header and sends the reply to the specified party. In a point-to-point configuration, a soft-phone that receives an ''INVITE'' message examines the ''Via'' header to determine the location of the sender. Then, using this information, it returns a ''100 Trying'' message. In the Unified Communications (UC) environment between a called and a caller SIP phone, there are the Session Manager, the Communication Manager and possibly the Session Border Controller (SBC). All SIP components use the ''Via'' header which is very simple to process.

There is also the ''Record-Route'' header. A UAS must copy all the ''Record-Route'' header field values from the request into the response, regardless of whether they are known to the UAS. A UAC may include a ''Route'' header field in a ''REGISTER'' request based on a preexisting route set. When a UAS responds to a request with a response that establishes a dialog (such as ''2xx'' to ''INVITE''), the UAS must copy all ''Record-Route'' header field values from the request message, including the URIs, URI parameters and ''Record-Route'' header field parameters, into the response message, regardless of whether they are known to the UAS. The UAS must maintain the order of these values and also add a ''Contact'' header field to its response message. This ''Via'' and ''Record-Route'' stacking allows a SIP request to pass through each agent, and each receiver of the message knows exactly how each subsequent reply will pass [43].

B. SIP REGISTRATION MECHANISM

SIP servers have a registration table. There are three different ways to register a SIP user. When a user authenticates for the first time, it reserves its place in the registration table for the default registration period of 1 hour and resend a registration message at the end of the default registration period [43]. When a SIP phone (hard-phone or soft-phone) is rebooted, it renews its registration, again reserves its place in the registration table for the duration of the default registration period and re-sends a registration message at the end of the default registration period. If a user whose registration is dropped, e.g. due to a network connection problem, attack, etc., wants to make a call, it first renews its registration by sending a registration message and reserves its place in the registration table for the default registration period. Depending on the SIP server's hardening settings and/or needs, using the multiple registration feature, a user can register from more than one IP address. A user that is not registered by the SIP server are called non-registered users.

In our attack implementation we select three different sets of users for both the caller and the callee: registered users, non-registered users and users randomly selected from registered/non-registered users. We implement and test our attack for both registered, non-registered and random users to create different server-side behavior and to see the server-side effects of these different operations.

C. SIP REQUEST BASED DISTRIBUTED REFLECTION DENIAL-OF-SERVICE ATTACK

For the proposed ''SR-DRDoS'' attack, our attack simulator misuses proxy servers to route SIP requests to a victim by including the victim's IP address in the ''Via'' and ''Record-Route'' headers of all SIP requests that it sends to reflectors. Any SIP component, such as the registrar, proxy server and SIP phone, can be used as a reflector.

In our attack, the attacker misuses SIP requests and sends them to SIP proxies which reflect them to the victim. The SIP proxy is tricked to reflect SIP requests to the victim by adding the victim's address to the ''Via'' header or the ''Record-Route'' header in the misused request message. Since the SIP request processing logic in a SIP server is complicated, the SR-DRDoS attack has a high potential for causing harm. For instance, when a SIP request is received from a SIP server, the server may have to deal with CPU-consuming operations, e.g. querying an SQL server to resolve the user's location, resolving an existing DNS or verifying

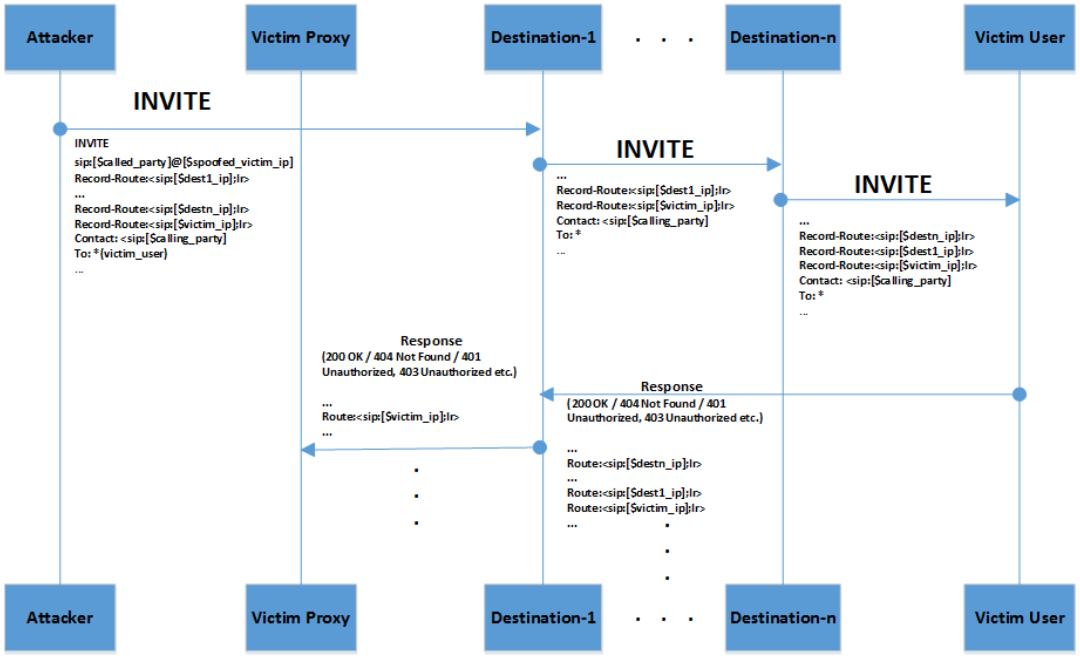

FIGURE 3. SR-DRDoS attack in detail.

MD5 credentials. An attacker can possibly be more evil and increase its attacking power further by using forking techniques [42], [43]. However, unless each one of the submitted requests is forked, a response of the same size is produced. Therefore, the amplification effect of such an attack is limited. One reason for using this type of an attack is to overcome possible firewall and NAT components that pass traffic only from known SIP proxies and drop other traffic [21].

As shown in the attack flow given in Figure 3, the "Via" and "Record-Route" headers are used in our SR-DRDoS attack. This makes destination machines forward "INVITE" messages to the victim machine and make it process all these requests, which requires too much CPU power due to the complex nature of SIP request processing. Hence, the victim machine is soon exhausted and the whole SIP network is affected.

D. ATTACK IMPLEMENTATION AND RESULTS

As our attack simulator, we use the SIP-DAS module of Mr. SIP,[1] and run it with the following command:

./mr.sip.py –ds –dm=invite –c 2 –di=172.16.215.130 –dp=5060 –r –to=toUser.txt —fu=fromUser.txt —ua=userAgent.txt —l

We observe the impact of our attack on 3 different user sets for both the caller and the callee: registered users, non-registered users and randomly selected users. When we apply our proposed SIP-based Request Reflection Attack, we observe that each INVITE message that is sent gets reflected. We observe that each transmitted INVITE message finds its way through reflectors and gets processed in the target machine. When we increase the number of reflectors, we observe that the number of packets increases proportionally. This makes destination machines forward INVITE messages to the victim machine. Thus, the victim machine processes all these requests which requires too much CPU power due to the complex nature of SIP request processing. Hence, the victim machine soon gets exhausted and the whole SIP network is affected.

The CPU usages of the SIP server for the registered, non-registered and random users during the first 4 minutes of the attack are given in Figure 4. When the attack is performed using random (registered and non-registered) users, the CPU load on the target server reaches 72% within the

TABLE 1. Mr. SIP tool commands list.

Short	Full	Description
–ds	–dos-simulator	Spoof IP addresses manually.
–dm	–dos-method	Select DoS packet type.
–c	–count	Set counter for how many messages to send. Default is Scapy.
–l	–lib	Use Socket library (no spoofing). If not specified, default is flood.
–di	–destination-ip	Set destination SIP server IP address.
–dp	–destination-port	Set destination SIP server port number. Default is 5060.
–r	–random	Spoof IP addresses randomly.
–to	–to-user	Set To User list file location. Default is toUser.txt.
–fu	–from-user	Set From User list file location. Default is fromUser.txt.
–ua	–user-agent	Set User Agent list file location. Default is userAgent.txt.

[1]The source code for Mr. SIP tool is published on https://github.com/meliht/mr.sip

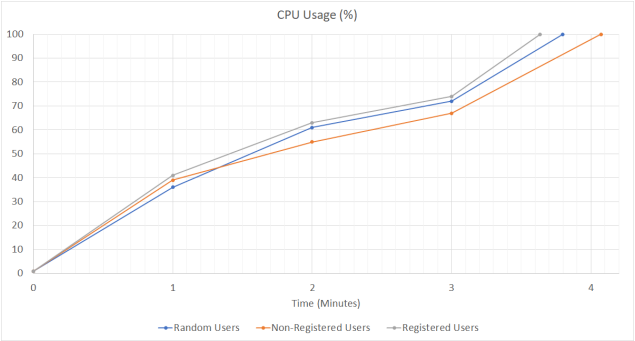

FIGURE 4. SIP server's CPU utilization after the attack is initialized.

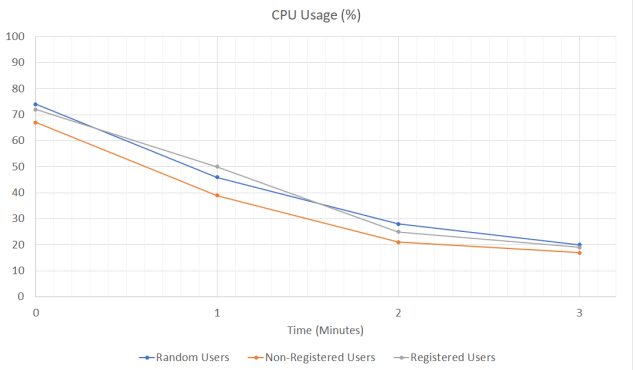

FIGURE 5. SIP server CPU usage during the 3 minutes after defense.

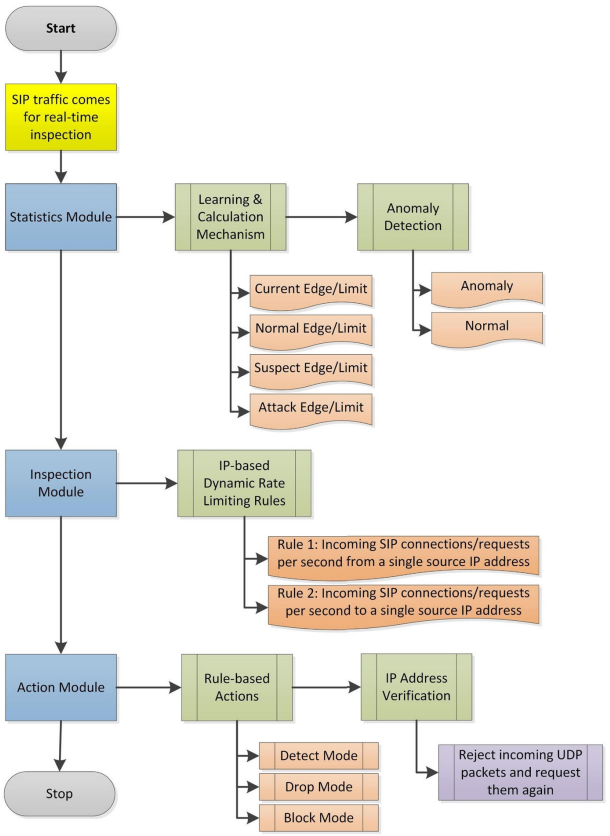

FIGURE 6. SIP DRDoS defense mechanism.

first 3 minutes of the attack and it reaches 100% in 3 minutes and 48 seconds. When the attack is performed with only non-registered users, the CPU load of the SIP server reaches 67% within the first 3 minutes of the attack and it reaches 100% in 4 minutes and 2 seconds. Finally, when the attack is performed with only registered users, the CPU load of the SIP server reaches 74% within the first 3 minutes of the attack and it reaches 100% in 3 minutes and 38 seconds. We would like to note that the SIP server in our laboratory environment has 0% initial CPU utilization when the attack is started. With the realization of our attack, in all three scenarios, the SIP server's CPU utilization reaches 100% and hence a complete loss of availability is observed around 4 minutes after the initialization of the attack.

During the attack, when a call from a user (registered or non-registered) arrives at the server, the server queries the registration table for both the caller and the callee. Depending on the size of the registration table, the response time of the query varies. Depending on whether the caller/callee exists in the registration table, i.e. whether she is registered or non-registered, the server behaves differently. In Figure 4, we observe a difference between the attack results for registered and non-registered users. We attribute this difference to the fact that the SIP server behaves differently to handle calls from registered and non-registered users. A call is established only when both the caller and the callee are registered users, therefore the number of handled calls are the highest when all users are registered users and lowest when all are non-registered users.

We would like to note that the defense mechanism given in [44], namely hop-count filtering, a commonly used IP Spoofing prevention technique, is active during our attack. Hence, hop-count filtering on its own does not seem to offer protection against our attack.

V. NOVEL DEFENSE MECHANISM

We propose a novel defense mechanism to mitigate the SIP-based DRDoS attack given in Sections IV-C and IV-D. Our defense mechanism is inspired by several techniques that are applied in other attack scenarios [1], [45], [50]–[55]. As shown in Figure 6, our defense mechanism has three modules, named as Statistics, Inspection and Action Modules.

Statistics Module collects windows of traffic periodically (hourly, daily, weekly, monthly) and for each window, it creates a sample traffic pattern by considering network and SIP packet specifications. This sample is named the normal traffic pattern. Firstly, the system owner/operator determines the VoIP network traffic period they want to sample. The selection may be hourly, daily, weekly or monthly, based on the intensity profile. If the intensity of VoIP traffic fluctuates according to the days of the week, i.e. if the system is busy on weekdays but idle at weekends, the system owner may want to sample the traffic on a daily basis. The traffic samples

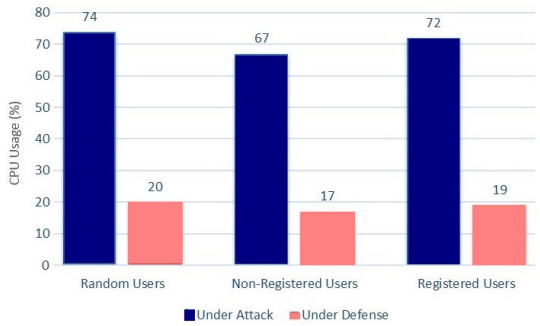

FIGURE 7. SIP server average CPU load for different user groups.

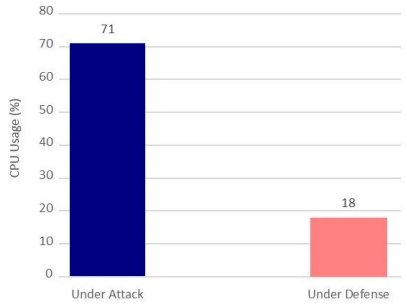

FIGURE 8. SIP server average CPU load for all users.

are then collected according to the determined time period. The lowest, the highest and the average threshold values are determined for each collected traffic sample. In order to calculate specific call parameters for the SIP DRDoS attack, the system collects the number of call initiation packets, established sessions, reflected requests and responses within the determined period.

Statistics Module has its own threshold calculator. By measuring bandwidth usages and packets per second for a time period, the learning mechanism calculates the attack traffic threshold. When the current traffic rate reaches the attack traffic threshold, which means there is anomaly, Inspection Module becomes active and compares the normal traffic pattern to the suspected attack traffic pattern. Note that similar techniques are used for anomaly detection in intrusion detection systems [52] and for detecting DDoS attacks [53]. By inspecting the headers/tags in SIP messages, e.g. Call-ID, from tag, branch tag, ''Record-Route'' header or ''Via'' header, it tries to identify how much of the suspicious traffic is auto-generated and should be dropped by Action Module in the defense mechanism. Inspection Module creates IP based dynamic rate limiting rules which are used by Action Module. In case of a suspected attack activity, Action Module puts the SIP server in one of the Detect, Drop or Block modes. If required, it may also use IP verification before dropping/blocking packets.

A. DEFENSE IMPLEMENTATION AND RESULTS

We implement our defense mechanism using our SIP-based defense tool named SIP-DD [2] with the following command parameters:

./sip-dd.py -d <device-name> -t
<inbound-traffic-limit-in-kbps> -v (verbose) -f <bpf-filter>

We use the pcap library in SIP-DD. It keeps a queue at the kernel level. Thus, it allows us to examine data asynchronously and perform detection after the actual traffic copy is received. For the required calculations, all rates are recorded in kilo bits per second (kbps). We select a calculation interval of $4-5$ seconds, considering performance, and implement multi-threaded rate control.

In order to test our defense mechanism, we first apply the SIP-based DRDoS attack introduced in Sections IV-C and IV-D. Before our defense mechanism is triggered and activated, the attack increases the CPU load of the SIP server to %74, %67 and %72, for random (both registered and non-registered), non-registered and registered users, respectively. In Figure 5, the CPU loads of the SIP server for random, non-registered and registered users are shown for the 3-minute time period after our defense mechanism is activated. We observe that the CPU loads for all user types decrease dramatically with our defense mechanism. We see in Figure 7 that the CPU load of the SIP server decreases from %74 to %20 for random users, from %67 to %17 for non-registered users and from %72 to %19 for registered users. Overall, the average CPU load decreases from %71 to %18, as given with Figure 8.

VI. CONCLUSION

We introduced a novel DRDoS attack, named as the SR-DRDoS attack, which exploits reflection based vulnerabilities in UDP based SIP signaling. Furthermore, we developed a novel attack tool, named Mr. SIP, and used it to realize our SR-DRDoS attack in a simulated version of an enterprise-grade SIP network. Our attack implementation was shown to dramatically increase the CPU load of a SIP server from %0 up to %100 within only 4 minutes after the initiation of the attack. Our SIP-Based DRDoS attack implementation proved to bypass the existing network defense mechanisms in SIP networks, such as firewalls, intrusion detection/prevention systems, black-lists, IP address based rate-limiting and packet-count based rate-limiting. Moreover, we proposed a novel defense mechanism that effectively mitigated the proposed DRDoS attack and proved more effective than existing defense mechanisms. While our attack implementation dramatically increased the CPU load of a SIP server up to %71, our defense mechanism was able to reduce the CPU load of the SIP server under attack from 71% down to 18%. We conclude that DoS derivative attacks can effectively be conducted against SIP and existing security mechanisms would be ineffective. Therefore, considering the widespread use of SIP technologies in sensitive institutional

[2]The source code for SIP-DD tool is published on https://github.com/meliht/sip-dd

and public infrastructures, it is crucial to increase research efforts on SIP security.

REFERENCES

[1] I. M. Tas, B. Ugurdogan, and S. Baktir, "Novel session initiation protocol-based distributed denial-of-service attacks and effective defense strategies," *Comput. Secur.*, vol. 63, pp. 29–44, Nov. 2016.

[2] T. Bessis, V. K. Gurbani, and A. Rana, "Session initiation protocol firewall for the IP multimedia subsystem core," *Bell Labs Tech. J.*, vol. 15, no. 4, pp. 169–187, Mar. 2011.

[3] Communications Fraud Control Association. (2017). *CFCA Fraud Loss Survey*. [Online]. Available: https://www.cfca.org/fraud-loss-survey

[4] D. Sisalem, J. Kuthan, and S. Ehlert, "Denial of service attacks targeting a SIP VoIP infrastructure: Attack scenarios and prevention mechanisms," *IEEE Netw.*, vol. 20, no. 5, pp. 26–31, Sep. 2006.

[5] J. J. Santanna, R. van Rijswijk-Deij, R. Hofstede, A. Sperotto, M. Wierbosch, L. Z. Granville, and A. Pras, "Booters—An analysis of DDoS-as-a-service attacks," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, Ottawa, ON, Canada, May 2015, pp. 243–251.

[6] T. Mahjabin, Y. Xiao, G. Sun, and W. Jiang, "A survey of distributed denial-of-service attack, prevention, and mitigation techniques," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 12, Dec. 2017, Art. no. 155014771774146.

[7] J. Stanek and L. Kencl, "SIPp-DD: SIP DDoS flood-attack simulation tool," in *Proc. 20th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Maui, HI, USA, Jul. 2011, pp. 1–7.

[8] M. Poongothai and M. Sathyakala, "Simulation and analysis of DDoS attacks," in *Proc. Int. Conf. Emerg. Trends Sci., Eng. Technol. (INCOSET)*, Tiruchirappalli, India, Dec. 2012, pp. 78–85.

[9] (2019). *8 Best DDoS Attack Tools (Free DDoS Tool Of The Year 2019)*. [Online]. Available: https://www.softwaretestinghelp.com/ddos-attack-tools

[10] (2019). *DDoS Simulations, Comsec*. [Online]. Available: https://comsecglobal.com/ddos-simulations-2/

[11] (2019). *Nimbus Attack Simulation Platform*. [Online]. Available: https://www.nimbusddos.com/ddos-testing.htm

[12] (2013). *Tools for Simulating DDoS Attacks*. [Online]. Available: https://serverfault.com/questions/515908/tools-for-simulating-ddos-attacks

[13] (2019). *MazeBolt's BaseLine Platform for DDoS Pentesting*. [Online]. Available: https://mazebolt.com/ddos-testing

[14] (2017). *DDoS Attack Simulation: Preparing For Large-Scale DDoS Attacks*. [Online]. Available: https://activereach.net/newsroom/blog/ddos-attack-simulation-preparing-for-large-scale-ddos-attacks

[15] (2019). *LoDDoS DDoS Test Platform*. [Online]. Available: https://www.loddos-sec.com/loddos-nedir

[16] (2019). *Simulating DDoS Attack Your Own Lab*. [Online]. Available: https://www.ixiacom.com/resources/simulating-ddos-attack-your-own-lab

[17] (2019). *Simulating DDoS Attacks With DDoSFlowGen*. [Online]. Available: https://galois.com/blog/2017/04/simulating-ddos-attacks-ddosflowgen

[18] A. B. Johnston, *SIP: Understanding the Session Initiation Protocol*. Norwood, MA, USA: Arthec House, 2009.

[19] E. Bou-Harb, M. Debbabi, and C. Assi, "Cyber scanning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1496–1519, 3rd Quart., 2014.

[20] R. K. C. Chang, "Defending against flooding-based distributed denial-of-service attacks: A tutorial," *IEEE Commun. Mag.*, vol. 40, no. 10, pp. 42–51, Oct. 2002.

[21] M. Voznak and J. Safarik, "DoS attacks targeting SIP server and improvements of robustness," *Int. J. Math. Comput. Simul.*, vol. 6, no. 1, pp. 177–184, Feb. 2012.

[22] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of network-based defense mechanisms countering the DoS and DDoS problems," *ACM Comput. Surv.*, vol. 39, no. 1, pp. 3–44, 2007.

[23] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: Classification and state-of-the-art," *Comput. Netw.*, vol. 44, no. 5, pp. 643–666, Apr. 2004.

[24] P. Mohana Priya, V. Akilandeswari, S. Mercy Shalinie, V. Lavanya, and M. S. Priya, "The protocol independent detection and classification (PIDC) system for DRDoS attack," in *Proc. Int. Conf. Recent Trends Inf. Technol.*, Chennai, India, Apr. 2014, pp. 1–7.

[25] D. R. Thomas, R. Clayton, and A. R. Beresford, "1000 days of UDP amplification DDoS attacks," in *Proc. APWG Symp. Electron. Crime Res. (eCrime)*, Phoenix, AZ, USA, Apr. 2017, pp. 79–84.

[26] V. Paxson, "An analysis of using reflectors for distributed denial-of-service attacks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 3, pp. 38–47, 2001.

[27] A. S. Tanenbaum and D. J. Wetherall, *Computer Networks*. Boston, MA, USA: Prentice-Hall, 2011.

[28] Y. A. Bekeneva and A. V. Shorov, "Simulation of DRDoS-attacks and protection systems against them," in *Proc. 20th IEEE Int. Conf. Soft Comput. Meas. (SCM)*, St. Petersburg, Russia, May 2017, pp. 165–167.

[29] D. Endler and M. Collier, *Hacking Exposed Unified Communications & VoIP Security Secrets & Solutions*. New York, NY, USA: McGraw-Hill, 2014.

[30] D. Sisalem, J. Floroiu, J. Kuthan, U. Abend, and H. Schulzrinne, *SIP Security*. Chippenham, U.K.: Wiley, 2009.

[31] W. Wei, F. Chen, Y. Xia, and G. Jin, "A rank correlation based detection against distributed reflection DoS attacks," *IEEE Commun. Lett.*, vol. 17, no. 1, pp. 173–175, Jan. 2013.

[32] F. Baker and P. Savola, *Ingress Filtering for Multihomed Networks*, document RFC3704, 2004. [Online]. Available: https://tools.ietf.org/html/rfc3704

[33] N. Athanasiades, R. Abler, J. Levine, H. Owen, and G. Riley, "Intrusion detection testing and benchmarking methodologies," in *Proc. 1st IEEE Int. Workshop Inf. Assurance (IWIAS)*, Darmstadt, Germany, 2003, pp. 63–72.

[34] N. J. Puketza, K. Zhang, M. Chung, B. Mukherjee, and R. A. Olsson, "A methodology for testing intrusion detection systems," *IEEE Trans. Softw. Eng.*, vol. 22, no. 10, pp. 719–729, Oct. 1996.

[35] (2017). *Zoiper SIP Client (Softphone)*. [Online]. Available: http://www.zoiper.com

[36] (2017). *Network Grep Network Packet Analyzer*. [Online]. Available: http://ngrep.sourceforge.net

[37] B. A. Sassani, C. Abarro, I. Pitton, C. Young, and F. Mehdipour, "Analysis of NTP DRDoS attacks' performance effects and mitigation techniques," in *Proc. 14th Annu. Conf. Privacy, Secur. Trust (PST)*, Auckland, New Zealand, Dec. 2016, pp. 421–427.

[38] B. Liu, S. Berg, J. Li, T. Wei, C. Zhang, and X. Han, "The store-and-flood distributed reflective denial of service attack," in *Proc. 23rd Int. Conf. Comput. Commun. Netw. (ICCCN)*, Shanghai, China, Aug. 2014, pp. 1–8.

[39] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Comput. Secur.*, vol. 28, nos. 1–2, pp. 18–28, Feb. 2009.

[40] B. Al-Duwairi and G. Manimaran, "Distributed packet pairing for reflector based DDoS attack mitigation," *Comput. Commun.*, vol. 29, no. 12, pp. 2269–2280, Aug. 2006.

[41] H. Beitollahi and G. Deconinck, "Analyzing well-known countermeasures against distributed denial of service attacks," *Comput. Commun.*, vol. 35, no. 11, pp. 1312–1332, Jun. 2012.

[42] D. Senie, *Changing the Default for Directed Broadcasts in Routers*, document RFC2644, 1999. [Online]. Available: https://tools.ietf.org/html/rfc2644

[43] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, *SIP: Session Initiation Protocol*, document RFC3261, 2002. [Online]. Available: https://tools.ietf.org/html/rfc3261

[44] H. Wang, C. Jin, and K. G. Shin, "Defense against spoofed IP traffic using hop-count filtering," *IEEE/ACM Trans. Netw.*, vol. 15, no. 1, pp. 40–53, Feb. 2007.

[45] M. Choraś, Ł. Saganowski, R. Renk, and W. Hołubowicz, "Statistical and signal-based network traffic recognition for anomaly detection," *Expert Syst.*, vol. 29, no. 3, pp. 232–245, Jul. 2012.

[46] B. Kurt, Ç. Yılız, T. Y. Ceritli, B. Sankur, and A. T. Cemgil, "A Bayesian change point model for detecting SIP-based DDoS attacks," *Digit. Signal Process.*, vol. 77, pp. 48–62, Jun. 2018.

[47] M. Semerci, A. T. Cemgil, and B. Sankur, "An intelligent cyber security system against DDoS attacks in SIP networks," *Comput. Netw.*, vol. 136, pp. 137–154, May 2018.

[48] J. Tang, Y. Cheng, Y. Hao, and W. Song, "SIP flooding attack detection with a multi-dimensional sketch design," *IEEE Trans. Depend. Sec. Comput.*, vol. 11, no. 6, pp. 582–595, Nov. 2014.

[49] J. Stanek and L. Kencl, "SIP protector: Defense architecture mitigating DDoS flood attacks against SIP servers," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 6733–6738.

[50] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Surveying port scans and their detection methodologies," *Comput. J.*, vol. 54, no. 10, pp. 1565—1581, 2011.
[51] P. Biddle, P. England, M. Peinado, and B. Willman, "The darknet and the future of content protection," in *Proc. ACM Workshop Digit. Rights Manage.*, 2002, pp. 155–176.
[52] L. Saganowski, M. Choras, R. Renk, and W. Holubowicz, "A novel signal-based approach to anomaly detection in IDS systems," in *Adaptive and Natural Computing Algorithms*. Berlin, Germany: Springer, 2009, pp. 527–536.
[53] L. Li and G. Lee, "DDoS attack detection and wavelets," in *Proc. 12th Int. Conf. Comput. Commun. Netw.*, 2003, pp. 421–427.
[54] W. Chen, Y. Liu, and Y. Guan, "Cardinality change-based early detection of large-scale cyber-attacks," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 1788–1796.
[55] H. Choi, H. Lee, and H. Kim, "Fast detection and visualization of network attacks on parallel coordinates," *Comput. Secur.*, vol. 28, no. 5, pp. 276–288, 2009.

I. MELIH TAS received the B.Sc. and M.Sc. degrees in computer science and engineering from Marmara University, Istanbul, Turkey, in 2007 and 2013, respectively. He is currently pursuing the Ph.D. degree in computer engineering with Bahçeşehir University, Istanbul.

He wrote the National VoIP/UC Security Standard Draft in cooperation with the Turkish Standards Institute. He is an Active Speaker in hacker conferences, including the Black Hat Arsenal, the Offzone, and the Nopcon. He holds an OSCP certificate and is currently working as a Principal Penetration Tester at Garanti Teknoloji, a subsidiary of Garanti BBVA Bank. His research interest includes the design and analysis of both offensive and defensive security mechanisms in the fields of VoIP security, network security, web application security and mobile application security.

BASAK GENCER UNSALVER received the B.Sc. degree in electrical engineering and the M.Sc. degree in control and automation from Yildiz Technical University, Istanbul, Turkey, in 2008 and 2011, respectively. She spent one year as a Visiting M.Sc. Student with the Technical University of Munich, Munich, Germany. She is currently pursuing the Ph.D. degree in computer engineering with Bahçeşehir University, Istanbul.

Between 2008 and 2017, she worked as a Telecommunications Software Support and Design Engineer at Nortel Netas, Istanbul, where she was involved in the company's cybersecurity research and development efforts. Since 2017, she has been working as a Cybersecurity Expert at Vodafone Turkey, with responsibilities in IT and mobile network security architecture consultancy, vulnerability and incident management, and software security testing. Her research interests include communication security, information security and privacy, web application security, and digital forensics.

SELCUK BAKTIR (Member, IEEE) received the B.Sc. degree in electrical engineering from Bilkent University, Ankara, Turkey, in 2001, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the Worcester Polytechnic Institute, Worcester, MA, USA, in 2003 and 2008, respectively.

He worked for companies, including the IBM T. J. Watson Research Center and Intel Corporation. In 2013, he founded the M.Sc. Program in cybersecurity at Bahçeşehir University, Istanbul, Turkey. His research interests include applied cryptography and computer security. He received the IBM Research Pat Goldberg Memorial Best Paper Award, in 2007; the European Union FP7 Marie Curie IRG Award, in 2010; and the TUBITAK Career Award, in 2016.