SECURING WEB SERVICES FROM DOS ATTACKS

Authors:
Bhagyashree Hemant Parkar
Parvesh Kopparapu

Abstract:

During the past few years, Web Services (WS) Technology has become the main reference architecture for heterogeneous systems. Companies nowadays are vulnerable to a lot of Web Services attacks as it is very important for them to use the internet for communication. There are many attacks that are practiced nowadays such as Denial of Services (DoS) attack, SQL injections, XML injections, XSS attacks, Xpath, and Spoofing which indeed makes it very difficult to secure confidential data stored on computers and servers while exchanging data during on operation over a network [1]. Among all these attacks, in this paper, we will be focusing on the Denial of Service attacks as it is the most commonly seen attack in Web Services. Distributed denial of service (DDoS) attacks are now everyday occurrences as your online services, email, websites, anything that faces the internet, can be slowed or completely stopped by a DDoS attack. The objective of this paper is to present a systematic review of the studies of web service security, attacks, and how to prevent them. It is identified that there is a lot of research going on in web services, dealing mostly with attack detection as well as identification of vulnerabilities in the services [3].

Keywords: Web services; Attacks; Security; Denial of service; Prevention

1. Introduction

We are able to communicate with each other on various independent platforms and/or languages only due to the Web Services that are available. Since the technology advanced, the net Services were challenged by different types of attacks. There is not any single solution for mitigating the attacks on Web Services although many solutions are proposed for minimizing these attacks[4]. The communication is performed using XML-based SOAP messages. Subsequently, the protection of a web services-based system relies on the security of the services themselves and also on the confidentiality and integrity of the XML-based SOAP messages utilized for communication [1]. DoS attacks often depend on malformed and/or overly long messages that engage a server in resource-consuming computations. For Web Services, an appropriate means to stop such forms of attacks is by conducting the total grammatical validation of messages by an application-level gateway before forwarding them to the server [3]. DOS attacks affect the provision of the system and its resources to valid requests. Recursive payload takes advantage of comprehensive XML tag nesting to overload the parsers thus causing XDOS (XML Denial-of-Service) attacks. A coercive payload loads a lengthy XML message into memory that utilizes an outsized number of CPU resources which eventually makes the server unavailable, contributing to DOS attacks [5].

2. Denial of Service Attack

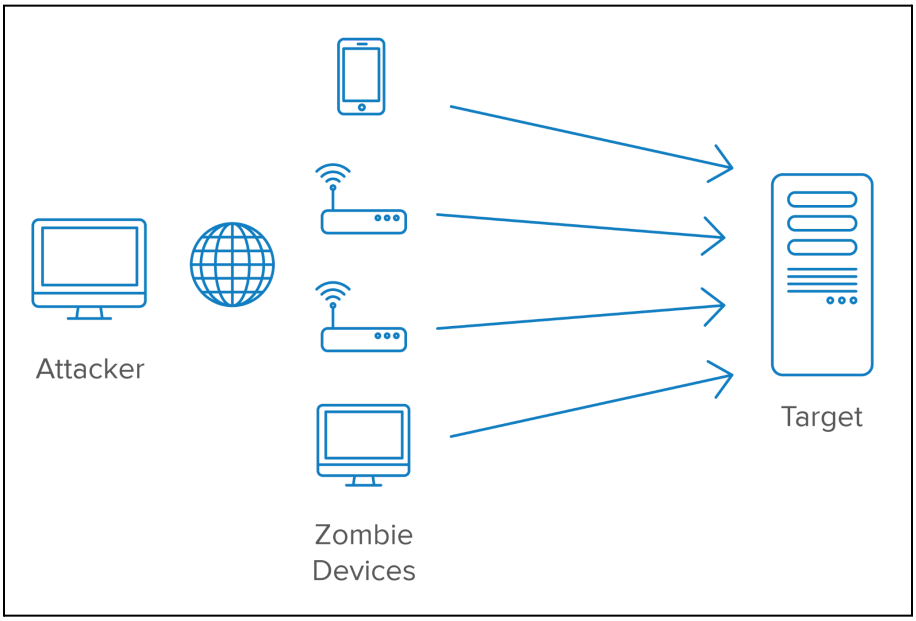
A Denial of Service (DoS) attack shuts down a machine or a network and makes it inaccessible to its intended users. By flooding the target with traffic or sending information which will trigger a crash is how DoS attacks work. In both instances, the DoS attack deprives legitimate users (i.e. members, employees, or account holders) of the service or resource they expected. Denial of Service (DoS) attack focuses mainly on reducing or completely eliminating a system or its availability. There are two types of DoS attacks:

2.1. Protocol Deviation Attacks

It exploits vulnerabilities within the implementations of protocol processing entities. In some cases, an attacked system is crashed just by diverging one packet from the intended protocol flow. A well known example is Ping of Death where an attacker attempts to crash, destabilize, or freeze the targeted computer or service by sending malformed or oversized packets employing a simple ping command [8].

2.2.Resource Exhaustion attack

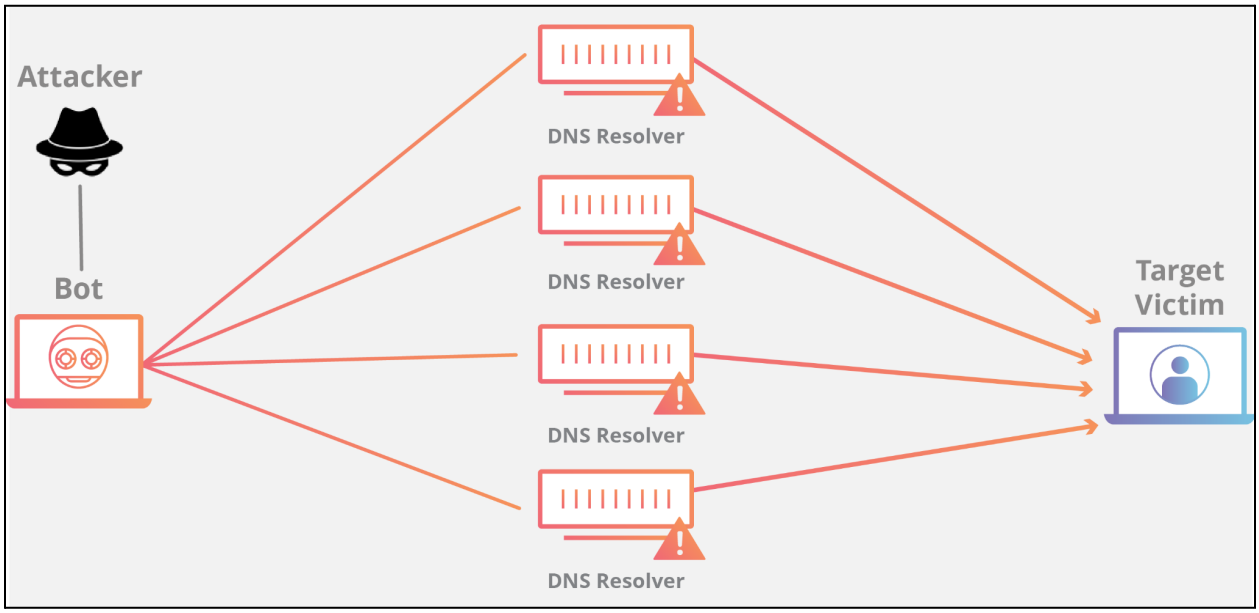
In this attack, the resources that are necessary to produce the service (i.e. network bandwidth, computation resources, and memory) are consumed. This leads to Dump Flooding as there is an extremely high network traffic load to the system that is providing the service. Even if executed as a Distributed Denial of Service (DDoS) attack, using such an attack makes it difficult to completely interrupt a service's availability. The attacker sends messages that are comparatively smaller in number but are suited to quickly exhaust the server's memory and CPU resources. A popular example is TCP/SYN Flooding, where the server is flooded with (small) TCP SYN packets and then it crashes due to memory consumption [3].



The above image explains a Denial of Service Attack [19].

3. Distributed Denial of Service Attack

An attack that uses multiple computers or machines to flood a targeted resource is a distributed denial-of-service (DDoS) attack. In a DDoS attack, there is a malicious attempt to disrupt the normal traffic of a targeted server, service, or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic. DDoS attacks achieve their effectiveness by utilizing multiple compromised computer systems as sources of heavy attack traffic. Computers and other networked resources such as IoT devices are mainly included in the exploited machines. A DDoS attack is like an unexpected traffic jam clogging up the highway, preventing regular traffic from arriving at its destination [17]. Cybercriminals have over the years developed a number of technical approaches for taking out online targets through DDoS.



The above image explains a DNS Amplification attack

The individual techniques tend to fall into three general types of DDoS attacks which are stated below.

3.1. Volumetric attacks

These attacks employ methods to generate massive volumes of traffic to completely saturate bandwidth, creating a traffic jam that makes it impossible for legitimate traffic to flow into or out of the targeted site [18].

3.2. Protocol attacks

They are designed to eat up the processing capacity of network infrastructure resources like servers, firewalls, and load balancers by targeting Network and Transport layer protocol communications with malicious connection requests [18].

3.3. Application attacks

Some of the more sophisticated DDoS attacks exploit weaknesses in the application layer by opening connections and initiating process and transaction requests that consume finite resources like disk space and available memory [18].

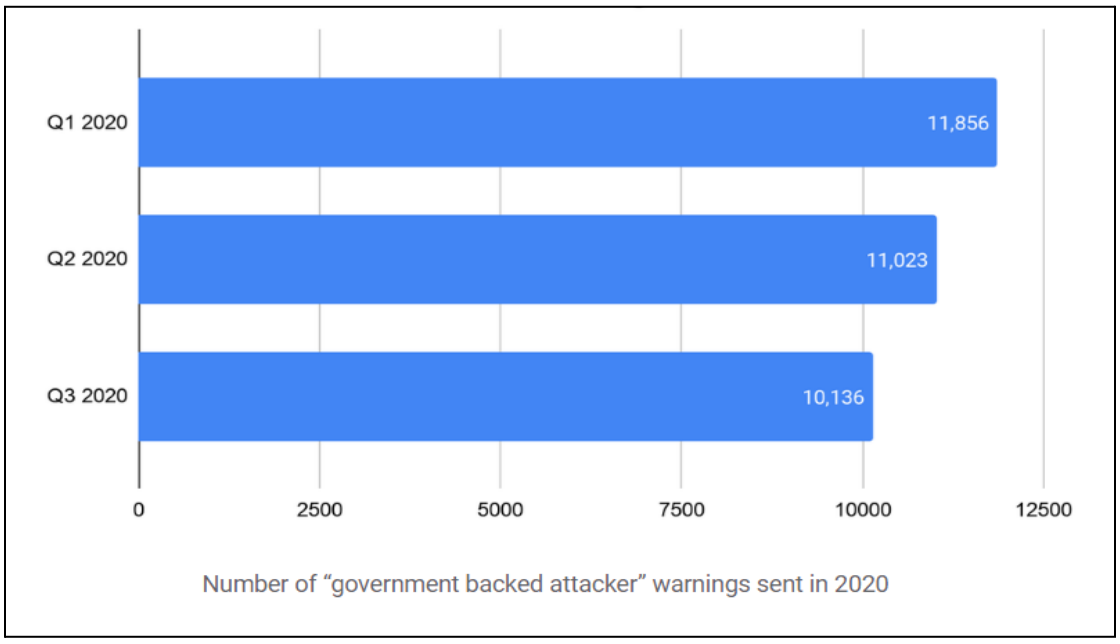
4. The Google Attack, 2020

In June 2020, TAG announced that they saw phishing attempts against the personal email accounts of the staff campaigners of Biden and Trump by Chinese and Iranian APTs (Advanced Persistent Threat) respectively. Although, they haven't seen such attempts being successful. The Iranian attacker Group (APT35) and the Chinese attacker group (APT31) targeted the campaign staffer's personal emails with credential phishing emails and they also added some tracking links in the emails. In their APT31 tracking activity, they have identified some deployed targeted malware campaigns [15].

One APT31 campaign was based on emailing links that would ultimately download malware hosted on GitHub. The malware was a python-based implant using dropbox for command and control. It allows the attacker to download and upload files as well as execute arbitrary commands. Every malicious piece of this attack was hosted on legitimate services making it even harder for defenders to rely on the network signals to detect. The attackers impersonated McAfee. The targets would be asked to install a legitimate version of McAfee anti-virus software from Github, while the malware was simultaneously silently installed to the system [15].

Above is the example prompt from an APT31 campaign impersonating McAfee

When they detect that a user is the target government-based attack, they send them a prominent warning. In these cases, they also share their findings with the campaign and the Federal Bureau of Investigation(FBI). This targeting is very consistent based on the previous reports [15].



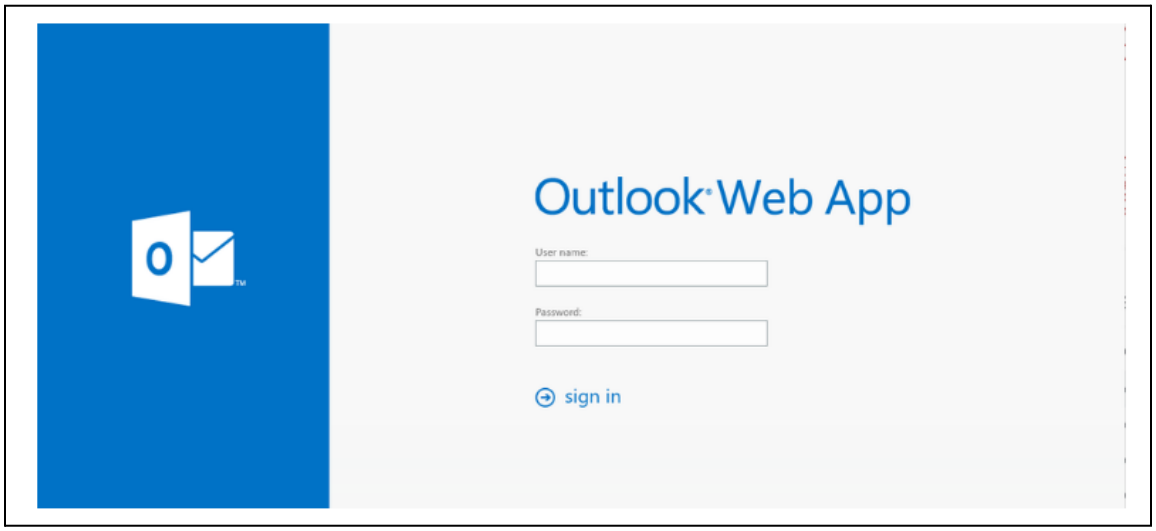
Government-Backed Attacker Warnings Sent in 2020

Overall, they've seen increased attention on the threats posed by APTs in the context of the US election. They've warned US government agencies about different threat actors and they've

worked very closely with the US government agencies and others in the tech industry to share their leads and intelligence about what they're seeing across the ecosystem. This has resulted in action on our platforms and on others as well. Then the US Treasury sanctioned Ukrainian Parliament member Andrii Derkach for attempting to influence the US electoral process, they removed 14 google accounts that were linked to him [15].

As the Covid 19 problems evolved, the threat actors improved their tactics too. The attackers targeted health organizations and they also tried to impersonate the World Health Organisation. In summer 2020, they observed threat attackers from China, Russia, and Iran targeting Pharmaceutical companies and researchers involved in vaccine development efforts[15]. In September 2020, they started seeing multiple North Korean attacker groups targeting Covid 19 researchers and pharmaceutical companies including those based in South Korea. One campaign used URL shorteners and impersonated the target's webmail portal in an attempt to harvest email credentials. In a separate campaign, attackers posed as recruiting professionals to lure targets into downloading malware [15].

In the threat actor toolkit, different types of attacks are used for different purposes: Phishing campaigns can be used as a scalpel targeting specific groups or individuals with personalized lures that are more likely to trick them into taking action (like clicking on a malware link), while DDoS attacks are more like a hatchet disrupting or blocking a site or a service entirely. While it's less common to see DDoS attacks rather than phishing or hacking campaigns coming from government-backed threat groups, we've seen bigger players increase their capabilities in launching large-scale attacks in recent years. Addressing state-sponsored DDoS attacks requires a coordinated response from the internet community, and we work with others to identify and dismantle infrastructure used to conduct attacks [15].



Spoofed Outlook login panel used by North Korean attackers attempting to harvest credentials

To ensure reliability, they have developed some innovative ways to defend against advanced attacks. Further in this paper, we'll dig deeper into DDoS threats, show trends, and explain how to prepare for multi-terabit attacks to keep our site alive and running.

4.1. Taxonomy of attacker capabilities

With a DDoS attack, an attacker wants to disrupt the victim's service with a flood of useless data traffic. This attack does not leak user data and does not cause compromise, but if not addressed promptly, it can lead to failure and loss of user trust.

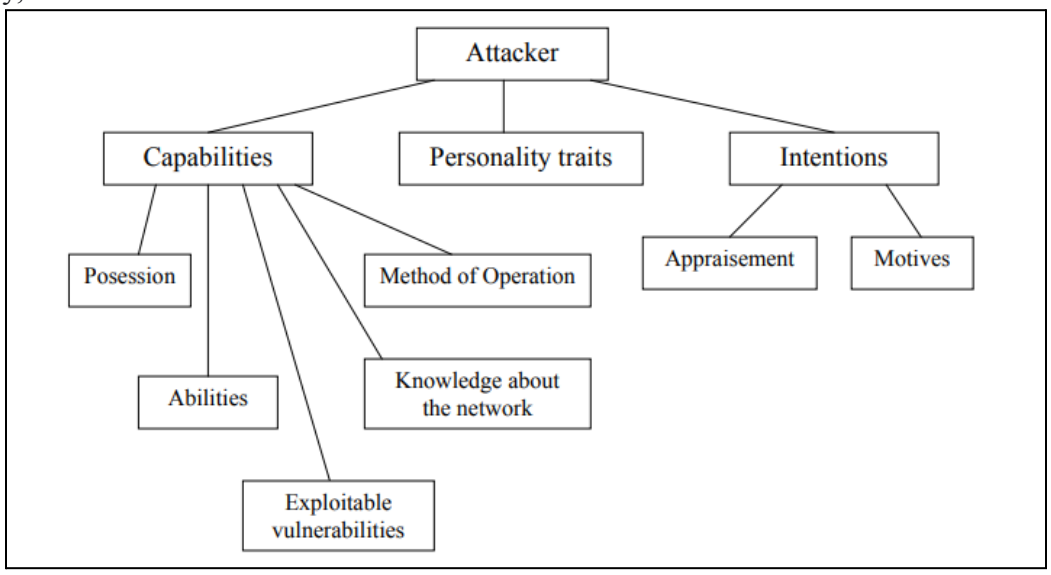


Fig: Characterizing an attacker

Attackers are constantly developing new technologies to disrupt the system. They give their attacks imaginary names like Smurf, Tsunami, XMAS Tree, HULK, Slowloris, Cache Bust, TCP Amplification, Javascript Injection, and 12 variations of reflection attacks. In the meantime, defenders need to consider all possible targets for DDoS attacks, from the network layer (routers/switches and link capacity) to the application layer (web, DNS, and mail servers). Some attacks may not even be able to focus on a specific target, but instead, target all IPs on the network. Multiplying dozens of attacks with different infrastructures to protect creates endless possibilities [20]. Rather than focusing on attack methods, Google groups volumetric attacks into a handful of key metrics.

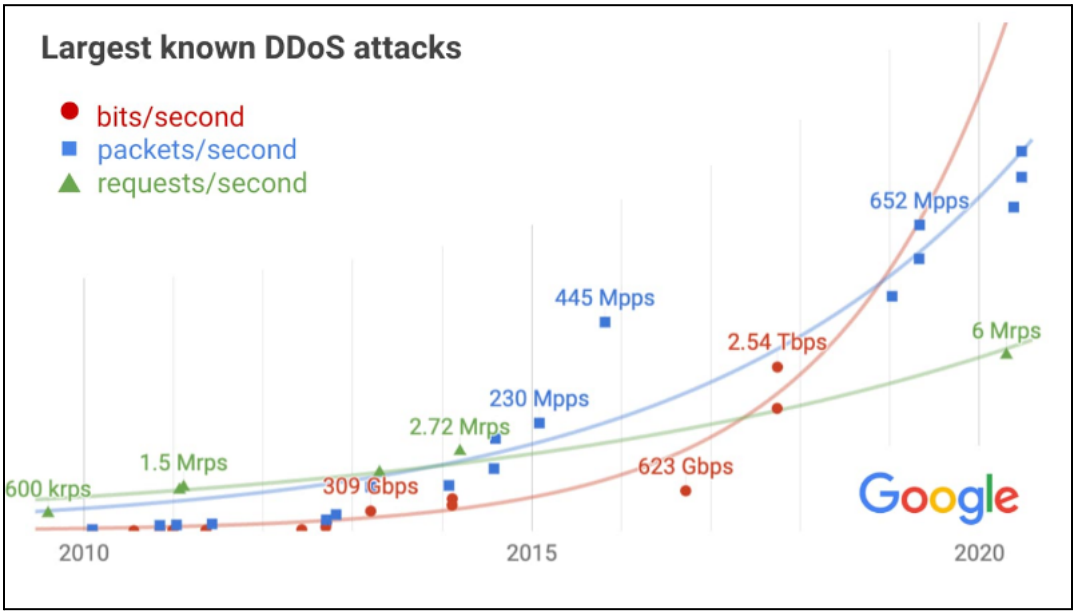
- **BPS** network bits per second is attacking target network links
- **PPS** network packets per second are attacking target network equipment or DNS servers
- **RPS** HTTPS requests per second are attacking target application servers

This way they can ensure that each system has sufficient capacity to withstand attacks measured by the relevant metrics.

4.2. Trends in DDoS Attack volumes

What they did here is that they determined the capacity needed to withstand the largest DDoS attacks for each key metric. This is a necessary step, getting this right is very essential for efficiently operating a reliable network - overprovisioning wastes costly resources, while underprovisioning can lead to an outage. To get this right, they have analyzed hundreds of significant attacks they received across the listed metrics and included credible reports by others.

They then plotted the largest attacks seen over the past decade to identify trends. Analyzing several years of data prior to this period has helped them in their decision of what to use for the first data point of each metric [20].



The image explains the Largest known DDoS attacks recorded by Google in 2020.

The exponential growth across all metrics is apparent, often generating alarmist headlines as attack volumes grow. But they needed to factor in the exponential growth of the internet itself which provides bandwidth and compute to defenders as well. After accounting for the expected growth, the results are less concerning, though still problematic [20].

4.3 Architecting Defendable Infrastructure

After observing the data and trends, they now extrapolated to determine the spare capacity needed to withhold the largest attacks likely.

1. BPS (network bits per second)

They created an infrastructure that absorbed a 2.5 Tbps DDoS in September 2017, the culmination of a six-month campaign that utilized multiple methods of attack. Despite simultaneously targeting thousands of Google's IPs, presumably in hopes of slipping past automated defenses, the attack had no impact. The attacker used several networks to spoof Mpps(millions of packets per second) to 180,000 exposed CLDAP, DNS, and SNMP servers, which would then send large responses to Google. They found out that this demonstrates the volumes a well-resourced attacker can achieve. This was four times larger than the record-breaking 623 Gbps attack from the Mirai botnet a year earlier. It remained the highest-bandwidth attack reported to date that reduced confidence in the extrapolation [20].

2. PPS (network packets per second)

They have observed a consistent growth trend, with a 690 Mpps attack generated by an IoT botnet in 2020. A notable outlier was a 2015 attack on a customer VM, in which an IoT botnet ramped up to 445 Mpps in 40 seconds—a volume so large they initially thought it was a monitoring glitch[20].

3. RPS (HTTP(S) requests per second

Google has observed that in March 2014, using a network man-in-the-middle attack malicious javascript injected into thousands of websites, caused hundreds of thousands of browsers to flood YouTube with requests, peaking at 2.7 Mrps (millions of requests per second). That was the largest attack known to google until recently when a Google Cloud customer was attacked with 6 Mrps. The slow growth is unlike the other metrics, suggesting they may be under-estimating the volume of future attacks [20].

Based on the previous observations Google estimated the expected size of future attacks, they prepared for the unexpected and therefore they over-provisioned their defenses accordingly. In addition, they designed their system to degrade gracefully in the event of overload and also wrote a playbook to guide manual responses if necessary. For example, their layered defense strategy allows them to block high-rps and high-pps attacks in the network layer before attackers reach the application servers. Graceful degradation applied at the network layer, too: Extensive peering and network ACLs were designed to throttle attack traffic which mitigated potential collateral damage in the unlikely event links became saturated [20].

4.4. Cloud-based Defenses

Google has noticed the scale of potential DDoS attacks can be devastating. Google deployed an armor called Google Cloud Armor integrated into the Cloud Load Balancing service which helps in absorbing massive DDoS attacks which can protect services in the Google Cloud, other Clouds, or on-premises from attacks. They recently announced Cloud Armor Managed Protection, which enables users to further simplify their deployments, manage costs, and reduce overall DDoS and application security risk [20].

Having sufficient capacity to absorb the largest attacks is just one part of a comprehensive DDoS mitigation strategy. In addition, they had to provide scalability, their load balancer terminates network connections on our global edge, only sending well-formed requests on to backend infrastructure. As a result, it can automatically filter many types of volumetric attacks. For example, UDP amplification attacks, syn floods, and some application-layer attacks will be silently dropped. In the next line of defense, they created the Cloud Armor WAF, which provides built-in rules for common attacks, plus the ability to deploy custom rules to drop abusive application layer requests using a broad set of HTTP semantics [20].

4.5. Working together for collective security

Google works with others in the internet community to identify and dismantle infrastructure used to conduct attacks. As a specific example, even though the 2.5 Tbps attack in 2017 didn't cause any impact, they reported thousands of vulnerable servers to their network providers and also worked with network providers to trace the source of the spoofed packets so they could be filtered. They encouraged everyone to join them in this effort. Individual users should ensure their computers and IoT devices are patched and secured. Businesses should report criminal activity, ask their network providers to trace the sources of spoofed attack traffic, and share information on attacks with the internet community in a way that doesn't provide timely feedback to the adversary. By working together, they said that we can reduce the impact of DDoS attacks [20].

5. The Intrusion Tolerant Approach

In order to detect low-rate DoS attacks, which exhaust the computational resources of the target machine for extended periods, an active resource monitoring approach is adopted. It consists in observing anomalous resource usage load (monitoring) and analyzing the possible causes of such resource usage overloading, in order to decide if such anomalous behavior is due to either an attack or a normal operation (diagnosis). When a DoS attack is detected, a threshold-based filtering reaction is triggered on the base of the target resource usage load (recovery).

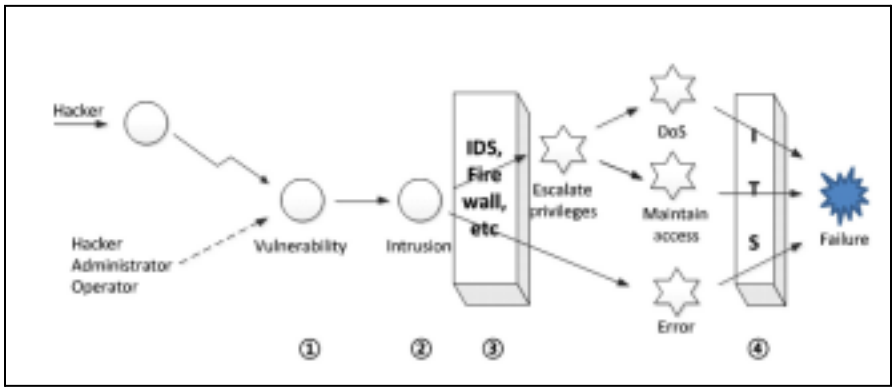


Fig.: The roles of intrusion-tolerant system (ITS). IDS: intrusion detection system, DoS: denial-of-service.

The CPU monitoring mechanism has to deal with application scenarios highly variable, which alternate periods of heavy workloads, which involve a high number of software components and heterogeneous types of service requests, with periods of low computational activities. We assume that during ‘normal’ operation, the monitored CPU behavior can be modeled as a random walk, which alternates stable periods, during which the CPU load has some ‘stable’ behavior (i.e., it is within a specific range), with transient periods (smaller compared with the stable), during which significant variation of CPU consumption occurs. During the transient period, changes in the monitored behavior consist in continuous increments or decrements with respect to the ‘stable’ behavior due to a workload variation (e.g., due to a burst of requests). Thus, a

count-and-threshold over-time monitoring mechanism using heuristic approaches is adopted to detect anomalous CPU consumption. It consists of monitoring extended excessive CPU consumption and detecting when a threshold is reached. The diagnosis activity consists to verify the presence of some stealth attack symptoms in the application requests [13].

It combines information collected by using different anomaly-based detection models that estimate the anomaly degree of monitored features, including: (i) the type of sequence of XML nested tags included in the message, (ii) the actual distribution of nested XML tags in the message sequence, and (iii) the number of nested XML tags of each message. Monitored symptoms are correlated (by means of a simple weighted sum) and rearranged based on a confidence level, which indicates the likelihood that they are symptomatic of an ongoing attack on the system. Finally, if the confidence exceeds a fixed threshold, a recovery action is performed [9,10]. The objective of the reaction is to reduce the CPU load on the target system, in order to reduce the period in which the service is unavailable. In particular, it filters each XML message that contains a number of nested tags greater than a given threshold TA. In order to face stealth attacks, we have to adapt the threshold, so that it cuts even low-rate attack messages (i.e., messages with a low number of nested tags). The ‘adaptive’ threshold TA is decreased until the CPU consumption falls below a severity level.

6. Our Proposed Solution:

In our opinion for the US election attack, the government agencies can ask their staff members not to download software or any application through the Open free Softwares System or by using any links which directs them to download any kind of application. In fact, they should create a mailing system especially for Professional purposes and not use it for Personal Use. They can ask the President to directly elect or assign trusted allies to monitor the emails used by the staff members. I know that there won't be any Privacy for the staff members but in order to maintain and run the Government organization they should not be any privacy inside the organization and it should not be accessible by outsiders at all, only for the staff members and you can even use Role-Based Access Control concept, by providing the president all the access controls like an owner and the next level officers providing access and rules according to their roles and level of confidentiality to prevent these kinds of attacks, and also you can add multiple login steps to provide more security. For example, when some staff member is trying to access some data by using login credentials, they can also add another step of authorization which is famously known protocol, The One-time password(OTP), and whenever a staff member is trying to login by using Login Credentials or OTP, the government agencies should record the process and monitor them carefully, verifying whether it's the authorized person or not trying to gain access. The pros of this solution are that it can be more secure with fewer resources but high maintenance, and the cons of this solution are that more computational resources, if the president is not authorized then the whole system will be compromised as he has all the access as an owner.

As for the Covid - 19 problem, we can use a similar solution, as the attackers were targeting pharmaceutical companies and researchers' login credentials. Even here they can use the Role-Based Access Control criteria to limit certain access to certain people. Based on the Level of Confidentiality access is given and also they can use the Signed Message-Digest concept as it

verifies the user who tries to access the service and also the service provider. Confirm whether the user asking to gain access is authorized or not. In this solution, the pros are again having multiple levels of verification makes it a little more secure, and the cons of this solution are again if the owner has access to everything then the system will be compromised.

For any solution, it cannot be completely secure, in order to prevent attacks and for the solution to be secured always try to detect attacks, monitor and search for patterns of the attackers, prevent the attack from happening, and if you detect any attack try to resolve it as soon as possible and make the service running and available for the users.

7. Conclusion

Web services have become the primary way in which key information is seamlessly exchanged between applications. This makes web service security an essential component and web service attacks are a serious threat to the integrity and availability of data. We have systematically analyzed papers on web service attacks. Most addressed attacks are Denial-of-Service attacks. Techniques to deal with attacks predominantly focus on attack detection measures. Since web service attacks cannot be completely eliminated, penetration and automation testing should be done as part of every development. This will guarantee added protection as well as lower attacks on web services.

8. References

[1] A Systematic Mapping Study on Web Services Security Threats, Vulnerabilities, and Countermeasures, 2021. Laila Bubaker, Aisha Yousef, Walid Algariani.

[2] Web Services Security (WS-Security) Version 1.0 05 April 2002. Bob Atkinson, Giovanni Della-Libera, Satoshi Hada, Maryann Hondo, Phillip Hallam-Baker, Johannes Klein, Brian LaMacchia, Paul Leach, John Manferdelli, Hiroshi Maruyama, Anthony Nadalin, Nataraj Nagaratnam, Hemma Prafullchandra, John Shewchuk, Dan Simon.

[3] Protecting Web Services from DoS Attacks by SOAP Message Validation. Nils Gruschka and Norbert Luttenberger

[4] Web Services Attacks and Security- A Systematic Literature Review. Varsha R. Mouli, K. P. Jevitha

[5] http://www.ws-attacks.org/Welcome_to_WS-Attacks WS-Attacks Welcome to WS-Attacks, 2015.

[6] https://www.a10networks.com/blog/5-most-famous-ddos-attacks/ Five Most Famous DDoS Attacks and Then Some

[7] Giinter Schafer. Sabotageangriffe auf Kommunikationsstrukturen: Angriffstechniken und

Abwehrmahahmen. PIK 28, pages 130-139,2005.

[8] https://www.imperva.com/learn/ddos/ping-of-death/ Ping of Death (POD)

[9] Ficco, M., Rak, M.: Intrusion Tolerant Approach for Denial of Service Attacks to Web Services. In: Proc. of the 1st International Conference on Data Compression, Communications and Processing (CCP 2011). IEEE CS Press (June 2011)

[10] Ficco, M., Romano, L.: A Correlation Approach to Intrusion Detection. In: Chatzimisios, P., Verikoukis, C., Santamar´ıa, I., Laddomada, M., Hoffmann, O. (eds.) MOBILIGHT 2010. LNICST, vol. 45, pp. 203–215. Springer, Heidelberg (2010)

[11] Zhang, Y., Mao, Z.M., Wang, J.: Low-Rate TCP-Targeted DoS Attack Disrupts Internet Routing. In: Proc. of the 14th Network and Distributed System Security Symposium, NDSS 2007 (February 2007)

[12] Li, Z., Wang, L., Chen, Y., Fu, Z.: Network-based and Attack-resilient Length Signature Generation for Zero-day Polymorphic Worms. In: Proc. of the IEEE Int. Conf. on Network Protocol, pp. 164–173. IEEE CS Press (October 2007)

[13] Intrusion Tolerance of Stealth DoS Attacks to Web Services. Massimo Ficco and Massimiliano Rak

[14] Lindstrom P (2004) Attacking and Defending Web Service. A Spire Research Report

[15] Hors AL, Hegaret PL, Wood L, Nicol G, Robie J, Champion M, Byrne S (2004) Document Object Model (DOM) Level 3 Core Specification. W3C Recommendation

[16] https://blog.google/threat-analysis-group/how-were-tackling-evolving-online-threats/ How we're tackling evolving online threats

[17] https://www.cloudflare.com/learning/ddos/what-is-a-ddos-attack/ What is a DDoS attack?

[18] https://cybersecurity.att.com/blogs/security-essentials/types-of-ddos-attacks-explained Types of DDoS attacks explained

[19] https://developer.okta.com/books/api-security/dos/what/ Types of Denial of Service Attacks

[20] https://cloud.google.com/blog/products/identifying-and-protecting-dos Exponential growth in DDoS attack volumes