Vulnerability Analysis and Management of an Internet Firewall

Chandana Middela, Anantha Dommeti, Kranthi Deekonda
Department of Information Systems
George Mason University, Fairfax, VA

Abstract

Firewalls protect a trusted network from an untrusted network by filtering traffic according to a specified security policy. A diverse set of firewalls is being used today based on the security policy. This paper initially examines a firewall in terms of its vulnerabilities concerning its operations. We examine one such vulnerability impact, Denial of Service (DoS) with respect to a most popularly known and used Cisco PIX (Private Internet Exchange) firewall which has been reportedly prone to attacks due to multiple software vulnerabilities. DoS attack occurs when a server does not provide services to genuine clients because it is too busy servicing the attacker client's requests. This happens because the attacking client continuously sends requests for service without completing the sessions. We analyze some of the reported problems and their causes. The paper addresses the importance of the vulnerability scores. The ability to score firewall vulnerabilities lays a foundation to prioritize the actions and response to the threat vulnerabilities present *In conclusion, this paper will present possible solutions and/or suggestions on how the vulnerabilities might be mitigated through some changes in configuring the firewall by enabling the unicast reverse path forwarding feature over the routing device which filters the spoofed packets through the best return path and then by introducing a path trace technique which would make use of encryption and tracing which complements the best path forwarding scheme and makes the spoofed packet filtering more effective.*

1. Introduction

Firewall is seen as crucial backbone in secure network infrastructures. Despite their critical modeling and testing to thrive in various potential attack conditions, firewalls have traditionally been exploited for several kinds of vulnerabilities. Vulnerability is defined as a bug, flaw, behavior, output, outcome or event within an application, system, device, or service that could lead to an implicit or explicit failure of confidentiality, integrity, or availability. A Firewall vulnerability is defined as an error or a flaw made during firewall design, implementation, or configuration that can be exploited to attack the trusted network. Because it is not feasible to examine each firewall separately for all potential problems, several researchers have generalized the vulnerabilities and introduced general mechanism for understanding firewall vulnerabilities in the context of firewall operations. A number of taxonomies [2], [3], [4], [5] provide high level descriptions of the classes of weaknesses that result in software vulnerabilities. We then use this analysis for our case study to answer some of our questions like Where does the vulnerability occur most often; what kind of effect do they usually have on the system; Which operations are traditionally more vulnerable to which kinds of errors and What kinds of attacks usually result from a particular type of error being found in a specific operation? Section 2 (2.0 – 2.2) deals with the related work and basics of vulnerability analysis and Section 3 examines the case study basics of the CISCO PIX firewall Architecture. The understanding of these is essential for further analysis. Section 4 introduces the problem statement and further analysis and section 5 and 6 deal with the proposed solutions and future work .

2. Related Work:
2.0 Firewall Vulnerability Analysis

To analyze and classify firewall vulnerabilities, we need vulnerability taxonomy and a scoring system that suits the analysis of firewall systems.

2.1 Taxonomy

Though there are several vulnerability taxonomies have been proposed in the literature, including [2], [5], [3], [4], [5]; Du andMathur's [3], [4] Software Vulnerability taxonomies are the most successful because they take into consideration all the other taxonomies and avoid some of the ambiguities in other taxonomies. It emphasizes on the point that each vulnerability has a cause, an effect on the system, and a fix that corrects the problem. We also found it more intuitive, and easier to work with in practice. For these reasons, we will use Du and Mathur's taxonomy in our analysis of vulnerabilities. We categorize each firewall vulnerability in the various Cisco PIX firewall versions according to causes, effects and fixes. Figure 1 [3], [4] gives us a Scheme for the three vulnerability categorization. Based on these categories we summarize the Du and Mathur taxonomy from [3], [4] of Cause, Impact and Fix in the context of an Internet firewall. From an operational viewpoint, any vulnerability is due to some reason, has an impact and may be fixed eventually.

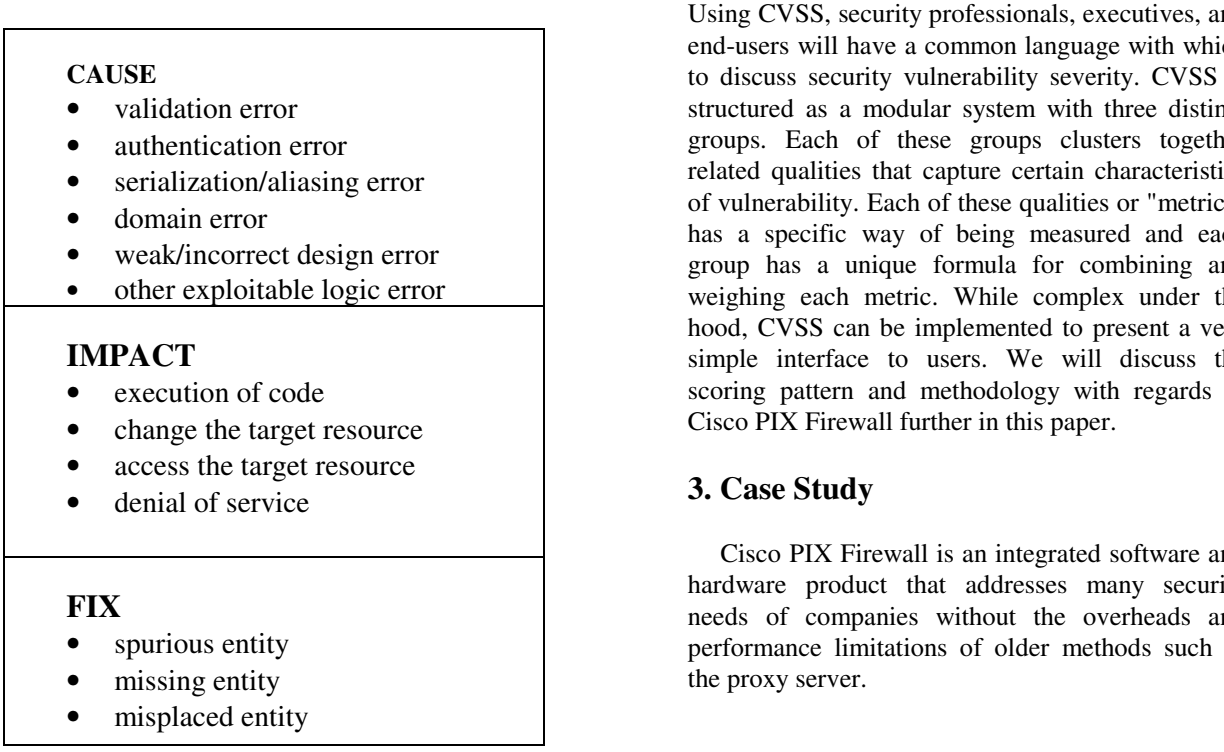


Figure1 [3],[4]Firewall Vulnerability Categories

2.2 Vulnerability Scoring System

The ability to score information system vulnerabilities lays a foundation to prioritize the actions and response to the threat vulnerabilities present. There are several competing, incompatible, and closed vulnerability scoring systems were the only available solutions. The lack of a unified standard in the space and resulted in much confusion when a single vulnerability would be released and would be scored differently among the different systems (sometimes resultant scores would be inversely correlated which made no sense).The Common Vulnerability Scoring System (CVSS)[7] is an open standard for scoring vulnerabilities. CVSS is used for scoring the vulnerabilities in the Cisco PIX firewall. CVSS is designed to rank information system vulnerabilities and provide the end user with a composite score representing the overall severity and risk the vulnerability presents. Using CVSS, security professionals, executives, and end-users will have a common language with which to discuss security vulnerability severity. CVSS is structured as a modular system with three distinct groups. Each of these groups clusters together related qualities that capture certain characteristics of vulnerability. Each of these qualities or "metrics" has a specific way of being measured and each group has a unique formula for combining and weighing each metric. While complex under the hood, CVSS can be implemented to present a very simple interface to users. We will discuss the scoring pattern and methodology with regards to Cisco PIX Firewall further in this paper.

3. Case Study

Cisco PIX Firewall is an integrated software and hardware product that addresses many security needs of companies without the overheads and performance limitations of older methods such as the proxy server.

It provides strong security functionalities with no adverse effect on network performance and is available in various models to meet the requirements of a range of networks. Cisco PIX Firewall provides complete protection by concealing the internal network from the internet. The network administrator also receives a complete account and logging of all transactions, including intrusion attempts on the internal network.

3.1 Features Of Cisco PIX Firewall

The PIX Firewall has succeeded in maintaining a simple, almost minimalist, list of components. It is one of the world's premier firewalls because its unique operation provides strong security and very high performance. The PIX Firewall uses The Adaptive Security Algorithm (ASA) and has its roots in Network Address Translation (NAT), with the ability to maintain information about the state of each connection that passes through it, and then filter (permit or deny) traffic based on that state. For this reason, it's classified as a stateful firewall.

3.1.0 Hardware and Software Components

The PIX Firewall series uses specially designed hardware and a very small, proprietary, multi-threaded kernel. PIX Firewall is easy to configure and hard to misconfigure. Unlike many firewalls, the PIX Firewall hardware and software are based on a pessimistic, or restrictive, security model. In other words, by default, everything is denied. To allow network traffic to pass through the PIX Firewall, it must be explicitly configured to accept that traffic. The PIX Firewall operating system itself is non-Unix, real-time and embedded. It isn't based on a mainstream operating system such as Windows or UNIX, but on a hardened, secure, embedded operating system known as Finesse.

3.1.1 PIX firewall Operation

Traffic moving through the network utilizes these three primary features:

- Cut-through proxy authentication mechanism that uses a security database such as TACACS+ or RADIUS to grant a network user or host access to an external network.
- Adaptive Security Algorithm (ASA) that monitors the traffic passing through the firewall. It also helps in providing stateful security by establishing and maintaining a state session table.
- Static translations and conduits, which allow access from external hosts to specific internal servers.

As we mentioned earlier, the PIX Firewall has its roots in NAT. Although it's possible to configure a PIX Firewall to not translate IP addresses, its switching process is based on NAT, and every packet must use this NAT mechanism.

Steps in NetworkAddressTranslation (NAT)

- A translation is a pair of IP addresses: local and global. The local address is on the network connected to the inside, or trusted, interface of the PIX Firewall. The global address is part of a network

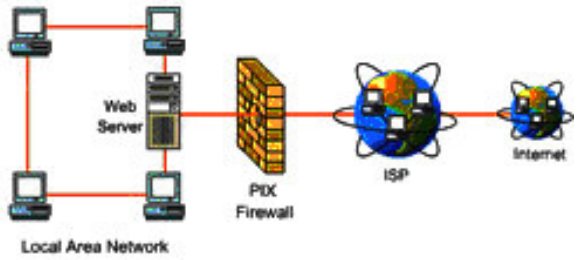


Figure 2: Static Translations and Conduits

somewhere beyond the outside interface that is trusted less than the inside interfaces. The PIX Firewall translates the local address to the global address as the packet passes outbound through the firewall. It translates the global address to the local address as a packet passes inbound through the firewall.

- Translations can be either static or dynamic. Static translations must be manually configured. Dynamic translations are created as packets that meet certain criteria.

- When the first packet in a series of packets arrives at the PIX Firewall from the inside interface, the PIX Firewall creates a translation slot. This "slot" is a software construct that keeps track of translations. Each translation uses one translation slot. Connection slots are another software construct that the PIX Firewall uses to keep track of stateful information.

- A given pair of devices, such as a client and server, can multiplex several conversations between their two IP addresses. This is often accomplished via TCP and UDP ports. For instance, a client could connect to a server via telnet, FTP, and HTTP simultaneously, creating three separate TCP connections between the two devices. If this happened across a PIX Firewall, it would create a single translation slot and three connection slots. Each connection slot is bound to a translation slot.

- The translation table, which is usually abbreviated as xlate table, is the actual table in memory that holds all the translation slots and connection slots. It's important to distinguish this table from the configuration file of the PIX Firewall. Just because you've configured a static entry doesn't mean it will appear in the output of the show xlate command. The PIX Firewall places an entry in this table only when a packet arrives.

- After a certain amount of inactivity (that is, after the PIX Firewall doesn't see any more packets that are part of this conversation), the PIX Firewall removes the entry from the xlate table. Remember that the xlate table shows the current translations and connections.

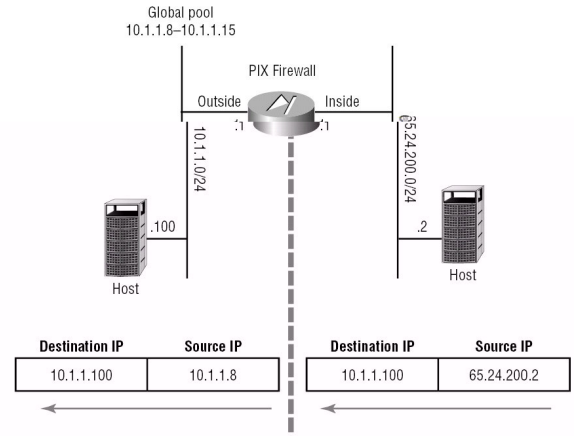
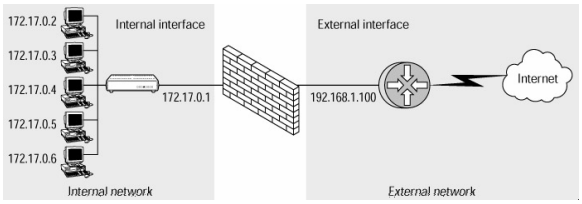


Figure 3 [8] Simple Network Address Translation

Port Address Translation

Port Address Translation (PAT) maps all internal IP addresses to one external global registered IP address. It is also called many-to-one address translation. In PAT, the firewall translates the IP address and port number of the internal host and adds an entry in the mapping table. The entry consists of the mapping of IP addresses and the port numbers. PIX Firewall uses this mapping table entry until the session is functioning. After the session terminates, the entry is deleted.



Packet Processing:

Outbound Packets:

When a packet arrives on the inside interface, the PIX Firewall first checks the xlate table for a translation slot. Specifically, this means the PIX Firewall checks the source address of the IP header and searches the xlate table for a match. Its next actions depend on whether it finds a match.

Packets with Existing Translation Slots

If the PIX Firewall finds a match for the outbound packet's source address, it knows it has seen packets from this address before and has already created a dynamic translation slot, or it has a manually configured static translation slot. The PIX Firewall then processes the outbound packet as follows:

1. It takes the global address from the translation slot that corresponds to the local address it just looked up in the xlate table and overwrites the source address in the IP header of the packet with the value of the global address.

2. The other attributes, such as the checksums, are recalculated. (Otherwise, the packet would be discarded upon arrival, since the change in the IP header would change the value of the checksum.)

3. The packet is forwarded out the outside interface.

Packets without Existing Translation Slots

If the PIX Firewall receives a packet on the inside interface that doesn't have a current translation slot in the xlate table, it can dynamically create an entry if it's configured to do so. In this case, when the packet arrives, the PIX Firewall checks the source address and finds no match in the xlate table. It then follows these steps to process the outbound packet:

1. The PIX Firewall makes sure it has sufficient connections, which are determined by the license.

2. It creates the translation slot by reserving an unused IP address from the global NAT pool and entering this global address along with the source address from the IP header into the translation slot.

3. With the translation slot created, the source address is overwritten with the global address.

4. The checksum and other values are recalculated.

5. The packet is transmitted on the outside interface.

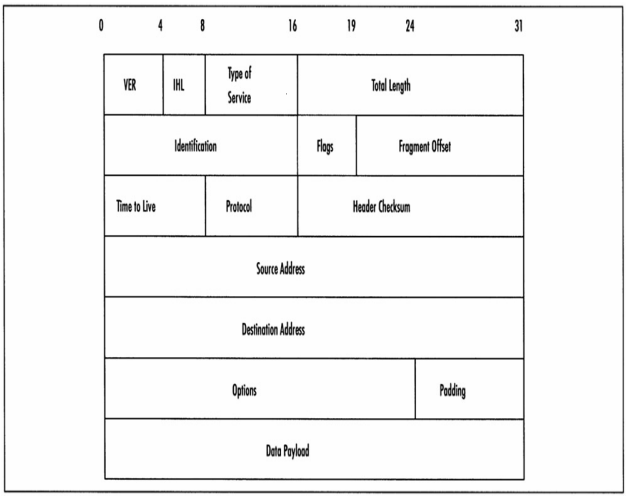


Figure 4 [9] Ip Header

Inbound Packets

For packets that arrive on the outside interface, destined for the inside network, the PIX Firewall behaves quite differently than it does for packets that arrive on the inside interface. This is because the outside network is less trusted.

- By default, packets from the outside don't create translation slots, so they can't be switched to the inside interface without a static NAT mapping. This makes the PIX Firewall very secure, from an architectural standpoint.
- But even before the PIX Firewall checks for an existing entry in the xlate table, packets from an outside interface must match criteria specified in an ACL. Only after an incoming packet matches the ACL will it be processed further.

The combination of the ACL and translation slot is the primary source of the PIX Firewall's security.

ROUTING:

From the description of packet processing in the previous sections, the PIX Firewall isn't a **router**. This is an important distinction, because many other brands (CISCO ISO) of firewalls are, in fact, routers, with packet-filtering or even stateful capabilities added on. In most cases, network is made up of a number of sub networks. There might be one or more at each branch office, or we might have the network configured so that each floor or each closet gets its own IP subnet. In any case, there are probably several internal routers. The hosts on most, if not all, of these subnets need access to external networks, such as the Internet. These internal and external networks are separated by the PIX Firewall, so routing on the firewall becomes an issue. Typically, the internal routers use an internal routing protocol, such as Open Shortest Path First (OSPF) or Enhanced Internal Gateway Routing Protocol (EIGRP), to communicate Network layer reachability information (NLRI). Presumably, in such a case, each internal network knows about all the other internal networks. They also typically use a default route for destinations they don't know about. In most internal networks, this default route points to the PIX Firewall, which means all traffic leaving the network goes through the PIX Firewall. The PIX Firewall, in turn, generally needs a default route to which it can deliver the packets that pass inspection

Adaptive Security Algorithm and Security Levels

Cisco's Adaptive Security Algorithm (ASA) is the basis for the PIX Firewall's security, and it includes much of the information discussed in the previous sections. However, it can be summarized into a few rules that govern how packets are inspected and permitted or denied:

- All packets must have a connection slot to be transmitted.
- All packets are allowed to travel from a more secure interface to a less secure interface unless specifically denied (for example, by an ACL).

- All packets from a less secure interface to a more secure interface are denied, unless specifically allowed.
- All ICMP packets are denied unless you specifically configure the PIX Firewall to accept them.
- When the PIX Firewall denies a packet, the packet is dropped (received but not transmitted), and the action is noted in the logs.
- Monitor returns packets to ensure that they're valid.

3.1.2. Security Features of PIX Firewall

These are some additional security features provided by PIX Firewall:

- **Flood Guard:** Controls AAA service's tolerance to half open login attempts. As a result, it prevents DoS attacks on AAA services and optimizes the use of the AAA system.

- **Flood Defender:** Protects the internal network from DoS attacks that flood an interface with TCP SYN packets.

- **FragGuard and Virtual Re-assembly:** Protects the network against IP fragmentation attacks.

- **URL Filtering:** When used with Net Partners Web sense product, PIX Firewall checks all outgoing URL requests with the policy defined on a Net Partners Web sense server, which runs either on Windows NT/2000 or UNIX. This server matches a URL request against a list of 17 Web site characteristics deemed inappropriate for business use. Based on its response, the PIX Firewall either permits or denies the connection.

- **Java Filtering:** Blocks Java applets from being downloaded into a protected network. Java applets are executable programs that are prohibited by some security policies because they may enable certain attacking methods on a protected network.

- **ActiveX Blocking:** Blocks HTML commands that specify the inclusion of ActiveX objects and comments them out of the HTML Web page. ActiveX objects might create potential problems such as causing workstations to fail or using network hosts to attack servers.

- **DNS Guard:** Identifies each outbound DNS resolve request, and allows only a single DNS response back. This usually happens when a host queries several servers for a DNS resolve request. After the first response to the request is allowed, additional responses are dropped by the firewall.

- **Mail Guard:** Provides safe access for Simple Mail Transfer Protocol (SMTP) connections from the outside to an inside e-mail server. It prevents a single mail server deployed within the internal network from being exposed to known security problems.

- **IPSec VPN:** Secures the transmission of sensitive data over unprotected networks. It provides secure communication tunnels between two peers, such as two PIX Firewall units.

- **PIX Failover**: Reduces downtime when a PIX firewall on a network breaks down. To enable this feature, you need two PIX Firewalls, one that performs the role of the primary firewall and the other that acts as its backup. If the primary firewall breaks down the secondary unit takes over the IP and MAC addresses of the primary unit and filters the data packets.

4. Problem Statement

In spite of so many security enhancing features Cisco PIX firewall versions contain multiple vulnerabilities.

- One such recently explored vulnerability was in the processing of IPSec IKE (Internet Key Exchange) messages. These vulnerabilities were identified by the University of Oulu Secure Programming Group (OUSPG) "PROTOS" Test Suite for IPSec which can be exploited by malicious people to cause **Denial of Service (DoS).**The vulnerability is caused due to errors in the processing of IKEv1 Phase 1 protocol exchange messages. This can be exploited to cause DoS via

specially crafted IKE packets. Cisco has assigned the following bug IDs :

1. Cisco PIX Firewall versions up to but not including 6.3(5) - CSCei14171
2. Cisco PIX Firewall up to but not including 7.0.1.4 - CSCei15053

Other Cases where spoofing was the effect of vulnerabilities in the firewall include:

On November 22, 2005, Symantec reported that a new denial-of-service vulnerability has been discovered which affects Cisco PIX firewall devices. Cisco PSIRT acknowledged the vulnerability.

This issue is being tracked by two Cisco Bug IDs:

- CSCsc14915 -- PIX 6.3 **Spoofed TCP SYN packets can block legitimate TCP connections**. This Bug ID tracks the issue for PIX software version 6.3 and older.
- CSCsc16014 -- PIX 7.0 **Spoofed TCP SYN Packets can block legitimate TCP Connections.** This Bug ID tracks the issue for PIX/ASA software version 7.0.

Cisco PIX firewall is vulnerable to a remote denial of service attack due to the way it routes TCP SYN packets with invalid TCP checksums. A full three-way handshake is not required in order for this to be successful. The attacker can therefore spoof the source IP addresses and ports. Normally these embryonic connections will time out in 2 minutes for versions up to and including PIX 6.3 or by default in 30 seconds for PIX 7.0. However if enough packets are sent, the attack may completely block all new traffic for the service resulting in a denial of service attack. These packets can be sent to all 65,535 possible ports or a port of the attackers choosing. It is important to note that the spoofed addresses can be external addresses so anti-spoofing techniques are not likely to be effective in blocking this attack. In addition, unlike the IP checksum, the TCP checksum is not normally verified as it traverses the network. Therefore these types of packets are not likely to get discarded before they reach an external PIX firewall nor will they be discarded by the PIX firewall. By sending a TCP SYN packet with an incorrect checksum through a PIX firewall, the PIX will block new TCP connections using the same source and destination TCP ports and IP addresses. Connections will remain blocked for approximately two minutes after which connections will be allowed. This behavior may be seen on all firewall interfaces but can be expected to have the most impact on TCP connections originating from higher security level interfaces to lower security level interfaces. Since the spoofed packets have an incorrect checksum, they are silently discarded by the destination and the firewall will not see a RST packet from either the destination or the legitimate source and will hold the embryonic connection open until the embryonic connection timeout which is 2 minutes by default. The root

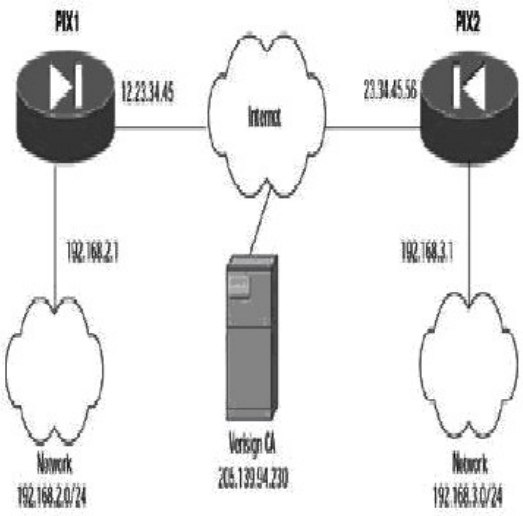


Figure 5[10] Network Set up for site-to-site VPN

cause is due to the spoofed packet creating an embryonic connection which sets up the TCP sliding window. A valid packet from a real host using the same connection as the spoofed packet sends a SYN over the same connection. The sequence number of the valid packet is out-of-window and rejected by the firewall's TCP sequence number check. Any subsequent retransmissions of the valid packet are also out-of-window and are rejected by TCP sequence number check. Other spoofed TCP SYN packets that create embryonic connections can also cause this behavior, blocking legitimate TCP connections until the embryonic connection times out.

Due to multiple occurrences of the DoS due to spoofed packets we address our research to this issue in this paper.

5. Analysis

We provide scores for the vulnerabilities based on the Common Vulnerability Scoring System (CVSS). Cisco will provide a base and temporal score. Customers can then compute environmental scores to assist in determining the impact of the vulnerability in individual networks. CVSS is standards based scoring method that conveys vulnerability severity and helps determine urgency and priority of response. Here we briefly describe how the Cisco arrives at the base and temporal scores [7] [14]. The Bug CSCsc14915 is due to spoofed TCP SYN packets**.** The CVSS overall vulnerability score is comprised of three distinct groups. Each of these groups clusters together related qualities that capture certain characteristics of vulnerability. Each of these qualities or "metrics" has a specific way of being measured and each group has a unique formula for combining and weighing each metric. [7] The metrics and the groups that comprise the score are briefly represented in the Figure 6.

5.1 Vulnerability score:

We are presenting Calculations the Vulnerability score for the Denial of Service attack as a result TCP SYN packet spoofing [14] for the Cisco pix firewall:

We use the CVSS [7] scoring system as mentioned above which involves three steps

5.1.1 Analyzing the Base metrics and calculating the Base Formula:

- Access Vector is "Remote" (It is remotely exploitable vulnerability that does not require authentication,)

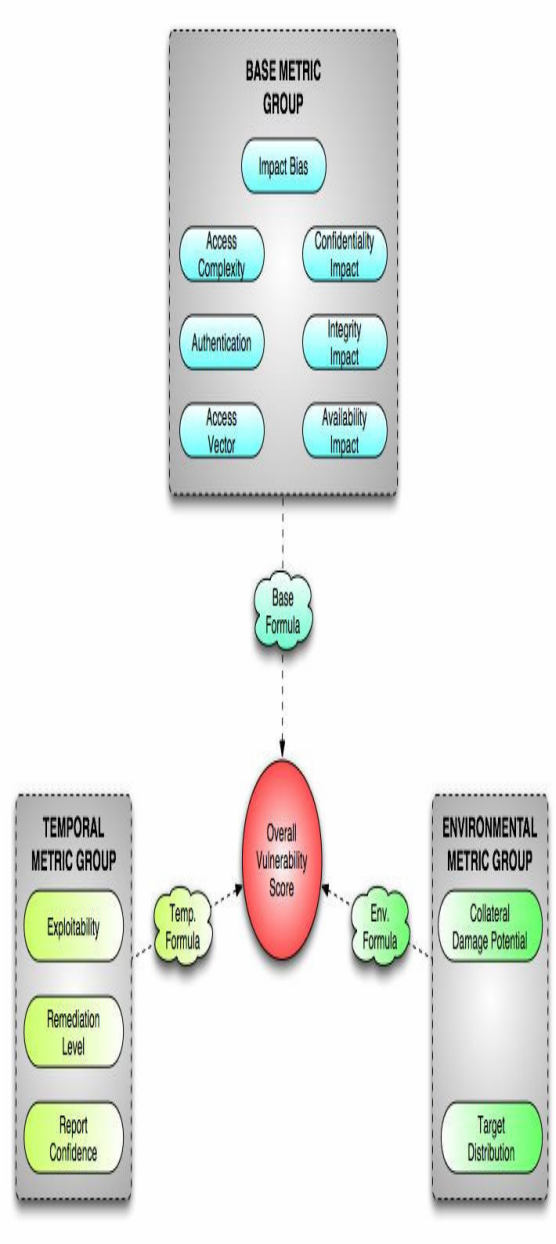


Figure 6 [7] Common Vulnerability Scoring System

- Authentication is "Not-Required".
- Access Complexity is "Low"(Because no additional access or specialized

- circumstances need exist for the exploit to be successful.)
- Integrity Impact metrics is set to "Complete" (because of the possibility of a complete system compromise.)
- Impact Bias is "Normal".

5.1.2 Analyzing the Temporal metrics and calculating the Temporal Formula:

- Known exploits do exist for this vulnerability and so Exploitability is "Functional".
- Cisco released a fix for this []the Remediation Level "Official-Fix" and Report Confidence "Confirmed".

Depending on the values for Collateral Damage Potential and Target Distribution the ENVIRONMENTAL (final) score could vary between 0.0 ("None", "None") and 6.3 ("High", "High"). Depending on the values for Collateral Damage Potential and Target Distribution the ENVIRONMENTAL (final) score could vary between 0.0 ("None", "None") and 6.35 ("High", "High". Depending upon the severity and the damage potential the organization can decide if it needs to take up immediate remedial measures.
The results are summarized below.

BASE METRIC	EVALUATION	SCORE
Access Vector	[Remote]	(1.00)
Access Complexity	[Low]	(1.00)
Authentication	[Not-Required]	(1.00)
Confidentiality Impact	[None]	(0.00)
Integrity Impact	[None]	(1.00)
Availability Impact	[Complete]	(0.00)
Impact Bias	**[Normal]**	**(0.333)**

Base Score = round_to_1_decimal (10
* AccessVector
* AccessComplexity
* Authentication
((ConfImpact ConfImpactBias)
+ (IntegImpact* IntegImpactBias)
+ (AvailImpact*AvailImpactBias)))

-----round (10 *(1.0 * 1.0 * 1.0 * (0.0) + (0.0) + (1.0 * 0.333))) = (3.3)

TEMPORAL METRIC | E VALUATION | SCORE

Exploitability	[Functional]	(0.95)
Remediation Level	[Official Fix]	(0.87)
Report Confidence	[confirmed]	(1.00)

TEMPORAL SCORE FORMULA

TemporalScore= round_to_1_decimal (BaseScore
* Exploitability
* RemediationLevel
* ReportConfidence)

---

round(3.3 * 0.95 * 0.87 * 1.00) == (2.7)

```
 ------------------------------------------------------------------
-----
 ENVIRONMENTAL
  METRIC                                EVALUATION
SCORE
 ------------------------------------------------------------------
-----
 Collateral Damage
     Potential              [None - High]         {0 - 0.5}

    Target
      Distribution          [None - High]         {0 -
1.0}
 ------------------------------------------------------------------
----

  ENVIRONMENTAL SCORE FORMULA
 EnvironmentalScore=round_to_1_decimal
                     ((TemporalScore +
                     ((10-TemporalScore)
                     * CollateralDamagePotential))
                       * TargetDistribution)
  ------------------------------------------------------------------
-----
round ((2.7 + ((10 - 2.7) * {0 - 0.5})) *
                     {0 - 1.00}) == (0.00 - 6.35)
------------------------------------------------------------------
-----
```

6. Solution

Cisco has suggested some Workarounds; methods sometimes used temporarily, for achieving a task or goal when the usual or planned method isn't working. The below is a discussion about the existing workarounds.

6.1 Disabling IKE feature:

IPSec is used in two general cases:

- The first case is LAN-to-LAN VPN operation in which two devices negotiates an IPSec connection between them for the purposes of connecting two remote LANs via an IPSec tunnel. In this case the devices negotiating the IPSec connection generally have static IP addresses, and the IPSec tunnel is up as long as there is traffic that needs to traverse the tunnel.
- The second case is a Remote Access (RA) VPN which is typically used to allow remote clients a connection to a secure network or service. A common example of this is a user connecting to a corporate network while away from the office. In this scenario, the remote user could be connecting from anywhere, and their IP address is not static, but rather dynamically assigned via the transport provider. The below discussed are solutions for the problem being discussed.

For customers that use IPSec, but do not require IKE for connection establishment, IPSec connection information may be able to be entered manually, and IKE can be disabled, eliminating the exposure. IKE is not a requirement for the establishment of IPSec connections. Depending on your requirements and the devices involved, it may be possible to statically configure the SA information and disable IKE. This type of configuration may not be possible in the case of RA VPNs due to the user's IP address being unknown prior to the establishment of the IPSec connection.

Another possible workaround is to mitigate the effects of this vulnerability by restricting the devices that can send IKE traffic to your IPSec devices. Due to the potential for IKE traffic to come from a spoofed source address, a combination of Access Control Lists (ACL's) and anti-spoofing mechanisms will be most effective.

6.1.2 Anti-spoofing by Unicast RPF

The Unicast RPF [13] [16]feature helps to mitigate problems that are caused by the introduction of malformed or forged (spoofed) IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. It is available on other Cisco routers and firewalls .A number of common types of denial-of-service (DoS) attacks can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks. For Internet service providers (ISPs) that provide public access, Unicast RPF deflects such attacks by forwarding only packets that have source addresses that are valid and consistent with the IP routing table. This action protects the network of the ISP, its customer, and the rest of the Internet.

When Unicast RPF is enabled on the Cisco PIX firewall interface, the router examines all packets received as input on that interface to make sure that the source address and source interface appear in the routing table and match the interface on which the

packet was received. This "look backwards" ability is available only when Cisco express forwarding (CEF) is enabled on the router, because the lookup relies on the presence of the Forwarding Information Base (FIB). CEF [16] generates the FIB as part of its operation. When a packet is received at the interface where Unicast RPF and access control lists (ACL's) have been configured, the following actions occur:

1. Input ACLs configured on the inbound interface are checked.

2. Unicast RPF checks to see if the packet has arrived on one of the best return paths to the source, which it does by doing a reverse lookup in the FIB table.

3. CEF table (FIB) lookup is carried out for packet forwarding.

4. Output ACLs are checked on the outbound interface.

5. The packet is forwarded.

Unicast RPF checks to see if any packet received at a router interface arrives on the best return path (return route) to the source of the packet. Unicast RPF does this by doing a reverse lookup in the CEF table. If the packet was received from one of the best reverse path routes, the packet is forwarded as normal. If there is no reverse path route on the same interface from which the packet was received, it might mean that the source address was modified. If Unicast RPF does not find a reverse path for the packet, the packet is dropped or forwarded, depending on whether an access control list (ACL) is specified in the IP verify unicast reverse-path interface configuration command.

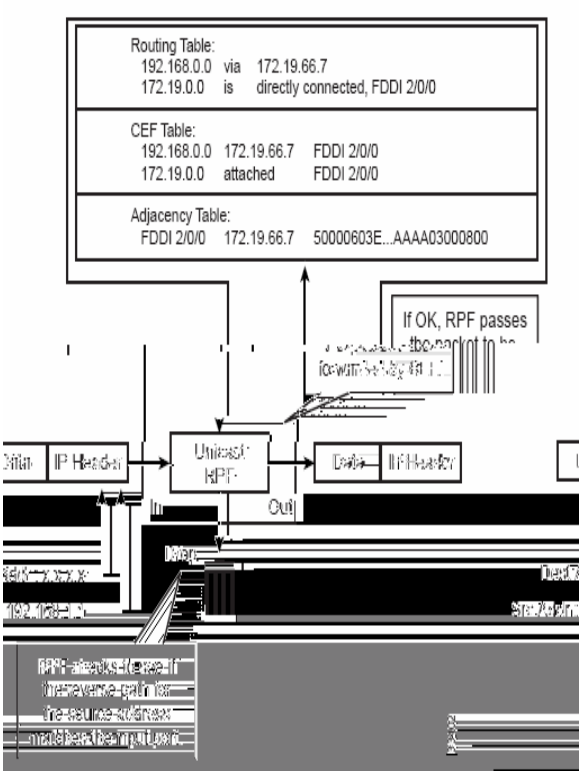


Figure 7 [16] Unicast RPF drops packets that fail validation.

6.1.2 Path Trace Method checking spoofed packets:

Defending against DoS attacks is extremely difficult because there is usually no explicit attack pattern to distinguish legitimate packets from malicious ones. Moreover, to hide the sources of attack traffic and circumvent DoS defense mechanisms relying on inspecting IP header fields, DoS attack programs generally fill IP header fields, especially the 32-bit source IP address, with randomized values. This IP spoofing technique has made the detection and filtering of DoS traffic extremely difficult, and it has become a common feature of the many DoS attack tools..

In this section, we propose spoofed packet filtering anti-DoS scheme which dis-allows packet with spoofed source address. It intends to complement, rather than replace the existing scheme. By weeding out spoofed IP packets constituting a dominant share of DoS attack traffic, DoS attackers are forced to use real source IP addresses in attack

packets. This allows packet filtering mechanisms to discard packets according to their source IP addresses.

Each IP packet traversing the network is embedded with a unique a trace or a flag(PathID) representing the route an IP packet has traversed and IP packets with incorrect PathID is considered spoofed. The basic principle of the scheme is the validation of an IP packet via its source IP address and the PathID embedded in it. The first step is computation of a PathID and then the inspection algorithm for identifying spoofed IP packets. Next, step would be modification of the SPTT (Source Path Trace Table) table of the PIX firewall by including the additional information that contains the mappings of Source IP addresses and their corresponding PathID. Coupled with the unicast RPF we would to propose a Path Trace method.

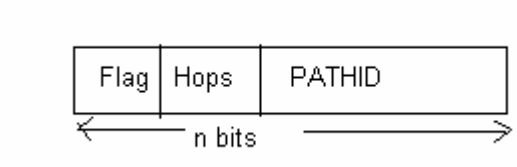


Figure 8: Path Trace

To generate a PathID representing the route an IP packet traversed, it is assumed that each participating router assigns each of its network interface an n-bit random number, and these random numbers are kept securely. These numbers should not be disclosed. Two fields, a d-bit hop field and n-bit path identification (PathID) field,[20] where the former represents the number of intermediate routers traversed, and the latter denotes an identifier derived from the random numbers associated with the traversed network interfaces in the route.

The path trace of an IP packet is stored in the IP packet header, and thus it is delivered to the destination host along with the packet. We also assume that a flag bit in the IP packet header is available for indicating the start of path trace. When a participating router receives an IP packet, it first examines the flag field. If it is 0, the receiving router is the first participating router the packet encountered in the path. In this case, the receiving router sets the flag bit to 1, sets the distance field to 1 and sets the PathID field to the random number associated with the incoming interface of the packet. On the other hand, if the flag bit is already 1, the receiving router increments the hop field by one, and updates the path identification field with H(PID, Nx), where PID represents the current value of path identification field in the packet, Nx denotes the random number of the incoming interface, and H is a hash function.

Packet traversal from the source S to the destination D across routers R1 to R4. The first router in the path, R1, sets both flag and distance filed to 1 and sets the initial PID value to the random number of the incoming interface, i.e. N1. Afterwards, each router increases the hop field and updates the PID field according to the previous PID value and the random number of the current incoming interface. In this figure, H denotes a hash function.

To allocate space from the IP packet header for storing a path trace, the 16-bit Identification field in the IP header is chosen to be overloaded. Issues related to the overloading of this field have been studied and reported [17]. In this paper, the 16-bit Identification field is divided into two sub-fields. The first sub-field is 5-bit long and is used to store the value of distance. It is believed that 5 bits are sufficient [18][19] since most of Internet paths are shorter than 31 hops. To deal with Internet paths with more than 31 hops, a simple solution is to use more bits. However, this will reduce remaining bits for storing path identification, and consequently increase the collision rates. To avoid increasing hash collisions, in our scheme, we choose to stop increment the distance field when its value reaches 31. Though in this case, Internet paths that have more than 31 routers supporting our scheme will have the same distance values; the path identification field can still help distinguish them if their path identifications are different. The remaining 11 bits of the path trace are used to store path identification. Finally, we propose to use the un-used bit of the FLAG field in IP header to store the value of the flag bit.

In this way, filtering of spoofed IP packets will be quite straightforward if the table that contains the mappings of IP addresses and their path trace. The source IP address and path trace of an IP packet are extracted from the IP packet header first. Next, by using the extracted source IP address, the correspondent path trace can be retrieved from the SPTT table. If the path trace of the given IP address cannot be found in the xlate table the packets are dropped .Otherwise, we compares the two path traces to allow the legitimate packets to traverse the network. To avoid errors caused by an obsolete SPTT table, spoofed packets will only be discarded after a

DoS attack signal is caught by employing certain DoS detection mechanisms.

6.1.4 Configuring and maintaining robust ACL

Although it is often difficult to block traffic transiting your network, it is possible to identify traffic which should never be allowed to target your infrastructure devices and block that traffic at the border of your network. Infrastructure ACL's are considered a network security best practice and should be considered as a long-term addition to good network security.

7. Summary and Future Work

In the proposed Path Trace Method server maintains for each of its communicating clients the mapping from the client's IP address to the corresponding path trace. The construction and renewal of these mappings is performed in an on-demand fashion that helps to reduce the cost of maintenance. To avoid errors caused by an obsolete SPTT table, spoofed packets will only be discarded after a DoS attack signal is caught which might be another subject for the future work.

The effectiveness of any of the above proposed solution is dependent on specific customer situations such as product mix, network topology, traffic behavior, and organizational mission. Due to the variety of affected products and releases, customers should consult with their service provider or support organization to ensure any applied workaround is the most appropriate for use in the intended network before it is deployed. No computer or computer network is completely secure. There are new vulnerabilities found or created everyday. The only way we as IT professionals can rest easy when we go home at night is to know that we are employing a minimal amount of security today and working towards more security tomorrow. However, all the hard work and money spent on the best security tools available doesn't do any good if users don't do minimal things such as securing passwords and locking down workstations when they leave at night.

10. References

[1] Multi-State Information Sharing and Analysis Center**;** http://www.msisac.org/advisories/2005/11_23.cfm

[2] Ivan Krsul, *Software Vulnerability Analysis*, Ph.D. thesis, Department of Computer Sciences, Purdue University, 1998, https://www.cerias.purdue.edu/techreports-ssl/public/98-09.pdf.

[3] Wenliang Du and Aditya P.Mathur, "Testing for software vulnerability using environment perturbation," in *Proceeding of the International Conference on Dependable Systems and Networks (DSN 2000), Workshop On Dependability Versus Malicious Faults*, June 2000, pp. 603–612, http://www.cerias.purdue.edu/homes/duw/research/paper/ftcs30 workshop.ps.

[4] Wenliang Du and Aditya P. Mathur, "Categorization of software errors that led to security breaches," in *Proceedings of the 21st National Information Systems Security Conference (NISSC'98)*, 1998, http://www.cerias.purdue.edu/homes/duw/research/paper/nissc98.ps.

[5] M. Bishop and D. Bailey, "A critical analysis of vulnerability taxonomies," in *Proceedings of the NIST Invitational Workshop on Vulnerabilities*, July 1996, also appears as Technical Report 96-11, Department of Computer Science, University of California at Davis (Sep.1996) at http://seclab.cs.ucdavis.edu/projects/vulnerabilities/scriv/ucd-ecs-96-11.ps. Also see"Classifying Vulnerabilities," "A Taxonomy of UNIX and Network Security Vulnerabilities" at http://seclab.cs.ucdavis.edu/projects/vulnerabilities/scriv/ucd-ecs-95-10.ps and"Vulnerabilities Analysis" by the same author

[6]SecurityAlertSeverityMatrix Published: MAY 19 2003http://www.microsoft.com/technet/security/alerts/matrix.mspx Symantec Threat Scoring System, CERT Vulnerability scoring.

[7] Mike Schiff man, Cisco CIAG, "A Complete Guide to the Common Vulnerability Scoring System (CVSS)" http://www.first.org/cvss/cvss-guide.html

[8] Wade Edwards et al. CCSP Complete Study Guide (642-501, 642-511, 642-521, 642-531, 642-541), Sybex © 2005

[9] Daniel Kligerman et al., Cisco PIX Firewalls: Configure, Manage, & Troubleshoot, Syngress Publishing©2005

[10] Rajesh Kumar Sharma et al, Cisco Security Bible, John Wiley & Sons © 2002

[11] *Cisco ISAKMP IKE Message Processing DoS* http://secunia.com/advisories/17553/

[12] Zachary Wilson, Hacking: The Basics

April 4, 2001,
http://www.sans.org/reading_room/whitepapers/hacke
rs/955.php?
[13] Configuring Unicast reverse path Forwarding
Copyright © 1992--2006 Cisco Systems, Inc. All
rights reserved.
http://www.cisco.com/univercd/cc/td/doc/product/soft