Global Information Assurance Certification Paper

GIAC Training and Certification
SANS 2001 Baltimore
GCFW Practical
Version 1.5e
Arkadi Karetnikov

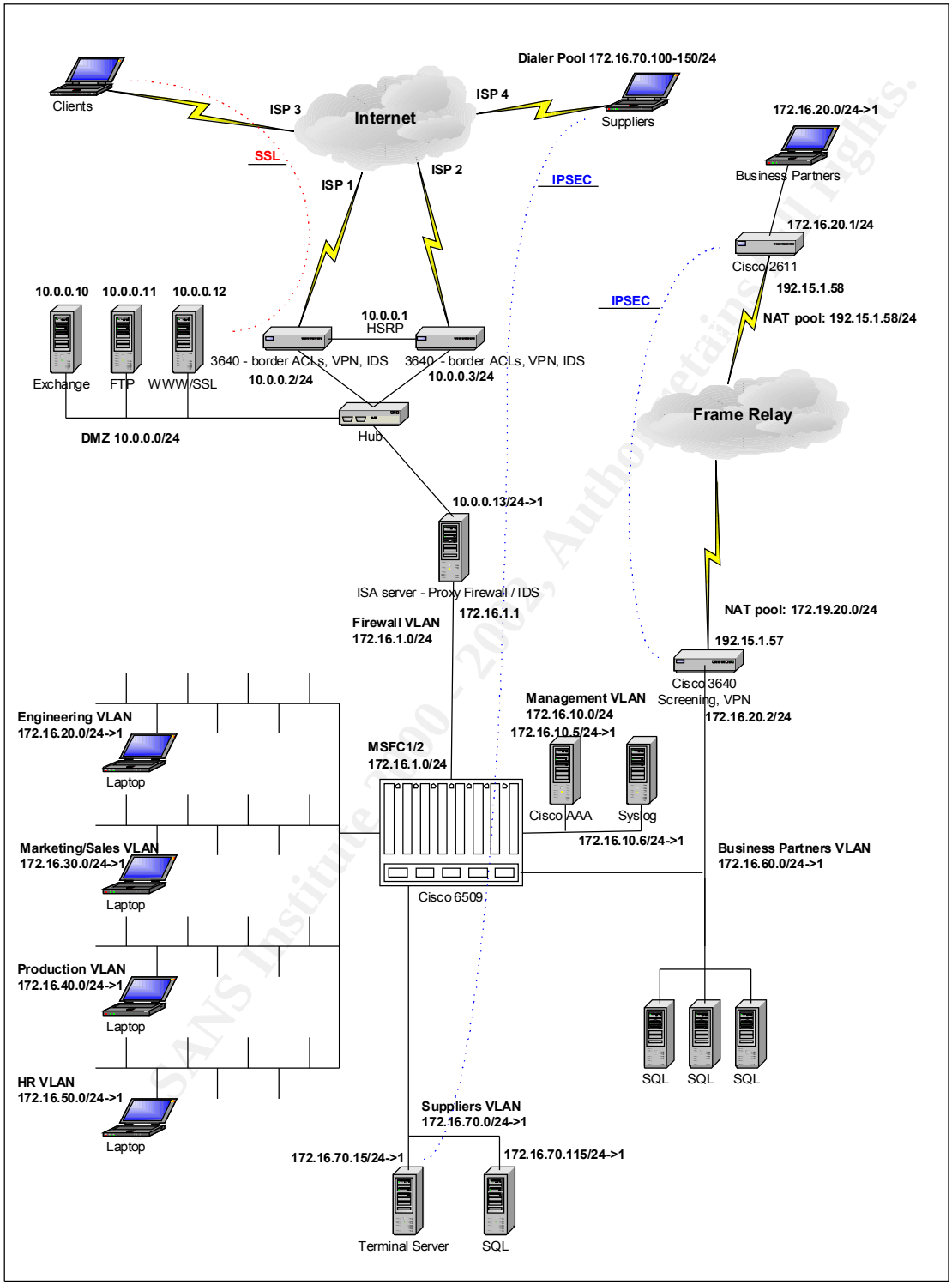
Assignment 1 - Security Architecture

Define a security architecture for GIAC Enterprises, a growing Internet startup that expects to earn $200 million per year in online sales of fortune cookie sayings, and which has just completed a merger/acquisition. Your architecture must specify filtering routers, firewalls, VPNs to partners, secure remote access, and internal firewalls. Be explicit about the brand and version of each perimeter defense component. Produce a diagram or set of diagrams with explanatory text that define how to use perimeter technologies to implement your security architecture.

You must consider and define access for:

- *Customers (the companies that purchase bulk online fortunes);*
- *Suppliers (the authors of fortune cookie sayings that connect to supply fortunes);*
- *Partners (the international partners that translate and resell fortunes).*

Below is the proposed architecture:



The border router consists of two physical boxes (Cisco 3640) running Cisco Hot Standby Router Protocol (HSRP) for the purpose of redundancy and recovery from DDOS attacks. Some actions were planned and done to minimize the impact of possible DDOS attacks, but in some cases the system will not be able to defend and will eventually go down. This design offers two T1 connections to two different ISPs, there is no load balancing, one link is in hot stand by state (passive) and another is active.

When a well planned DDOS attack happens and eventually brings the primary link down, (for instance, the active router crashes) the HSRP will activate the secondary link to another ISP. No reconfiguration of the internal hosts will be required as both routers share one virtual IP address, and the internal hosts in fact refer to the virtual IP address as their default gateway.

If the active router manages to stay on-line, but the link is fully saturated, the switchover may be done manually by simply turning off the active (under attack) router, HSRP will take care of the rest.

The border router(s) (logically it's one unit) will also act as a VPN gateway for remote VPN users (Suppliers), accessing the Terminal Server, which in turn, runs a customized database application. MS Terminal Server will allow low speed (dial up) clients to access and update the SQL database.

The border routers will also perform some context-based access control functionality (Cisco IOS CBAC), and NIDS (Network Intrusion Detection System). Publicly reachable hosts (mail, ftp, www) have some protection against TCP SYN floods by using of half-open connection limits and other IDS features on the border router.

As we are not running dynamic routing protocols on the border router, and using static routes only, the Cisco 3640 with 64MB of DRAM is perfectly capable of performing the above functionality. If at any stage the router performance becomes an issue, a separate hardware VPN encryption module can be installed in the existing chassis.

It was decided that the ISP would be responsible for managing the DNS server, so in our DMZ we only have the Exchange, FTP and WWW (SSL for on-line transactions) servers. All the addresses are private; the NAT (in fact, PAT) process on the border router will translate http, ftp and smtp requests into respective private addresses on the 10.0.0.0/24 network.

The second line of defense is the proxy firewall – MS Internet Security and Acceleration server (ISA server) which acts as a multilayer firewall with packet-, circuit-, and application-level traffic screening, stateful inspection, system hardening, and intrusion detection. It also performs one more level of network address translation (10.0.0.0/24 ->172.16.0.0/16). Internet access policies are implemented using the ISA policy manager.

The core switch of the internal network is implemented using a Cisco Catalysts 6509 switch with dual MultiLayer Switch Feature Card (MSFC), used for inter-VLAN routing

and redundancy. The Layer 3 switching provides a line of defense and prevention against some internally originated attacks.

There are eight virtual LANs:

- Firewall VLAN (172.16.1.0/24) – connects redundant MSFCs to the ISA server.
- Management VLAN (172.16.100.0/24) – this VLAN is used for system management purposes only, no workstations allowed to join that VLAN.
- All the internal users are divided into several groups, representing Engineering VLAN (172.16.20.0/24), Marketing/Sales VLAN (172.60.30.0/24), Production VLAN (172.16.40.0/24) and HR VLAN (172.16.50.0/24) respectively. As we use MSFC for inter-VLAN routing, and the MSFC considers each VLAN as a directly connected network, there is no need for a dynamic routing protocol or static routing entries in order to route between the VLANs, in contrary, special care needs to be taken to isolate the VLANs. That can be done by creating and applying ACLs to the virtual interfaces, associated with their respective VLANs. In that case, the MSFC will act as an internal packet filtering firewall. This will be covered in detail in assignment # 2.
- There are two other VLANs – Suppliers VLANs (172.16.70.0/24) and Business Partners VLAN (172.16.60.0/24). Those will have access only to their respective networks and will be denied access to any other VLANs.

We are using a private IP address range (172.16.0.0/16) for our internal network. Surprisingly enough, our business partner also uses the same addressing scheme, so there are some additional steps that need to be done to allow IP connectivity between overlapping networks. That's covered in detail in the next part of the assignment. We will be using Frame Relay connections and IPSEC VPNs to allow our business partners access confidential information stored on our SQL database server farm on the Business Partners VLAN. To minimize costs and management overhead, (yes that's right, we don't have a PKI in place, and running & supporting a dedicated CA would be impractical in our case) shared secret is a method for establishing IPSEC tunnels between the Frame Relay sites. Again, a Cisco 2611 router will be used as an internal firewall and a VPN site-to-site gateway.

Suppliers will have access to the internal SQL server via dial-up VPN connections. Cisco AAA server will be used as a TACCAC+ server to authenticate dial up users and proceed with IPSEC tunnels establishment.

A dedicated syslog server 172.16.10.6 located on the Management VLAN (Kiwi's syslog daemon running on W2K (sorry it's a Windows house) will collect all the syslog messages from the Cisco equipment.

Assignment 2 - Security Policy

Based on the security architecture that you defined in Assignment 1, provide a security policy for AT LEAST the following three components:

- *Border Router*
- *Primary Firewall*
- *VPN*

Security policy for the enterprise is as follows:

Internet access.

- Internet access is allowed for users on Engineering, Marketing/Sales, HR VLANs during and after business hours (8am – 5pm)
- Internet access is denied for Business Partners and Suppliers VLANs at all times
- Internet access is denied for users on Production VLAN during business hours and allowed after hours (5pm – 8am)

Inter VLAN access.

- No VLAN is allowed access to the Management VLAN
- Engineering, Marketing/Sales, Production, HR, Suppliers, and Business Partners VLANs have no access to any other VLANs, except for their own and Firewall VLANs.

Border router/firewall.

- Any traffic, originated from inside the network, is allowed to go out.
- No traffic, originated from outside (the Internet) is allowed inside, with two exceptions:

  Return traffic (established sessions)

  SMTP, HTTP, SSL, FTP requests to the DMZ

Communication with Suppliers

- All traffic between the Suppliers VLAN and remote Suppliers clients will be encrypted using 3DES

Communication with Partners

- All traffic between the Business Partner VLAN and remote Business Partner network will be encrypted using 3DES

- The only network that can be accessed by Business Partners is the Business Partners VLAN.

Content filtering

- No content filters will be applied at this stage. However, Internet traffic from all the internal VLANs should be monitored and logged by the ISA server and regularly reviewed.

Okay we'll start with the border router.

First, all unnecessary TCP and UDP services need to be switched off. This included echo, chargen, and discard – if left enabled, they can be used to launch DOS attacks:

no service tcp-small-servers
no service udp-small-servers

Next – Cisco CDP – Cisco Discovery Protocol. While it may be useful for diagnostic and troubleshooting purposes, it should be turned off on any router that's directly reachable from the Internet:

no cdp enable

Network Time Protocol. In some cases, especially in large enterprise networks, running NTP may be useful to synchronize time, but in our case we don't use it so we'll disable NTP:

no ntp enable

We well also disable SNMP and other things since we don't use it

no snmp server
no ip http server

Router access

It's considered to be good practice not to have telnet access configured on the router, which acts as a firewall. On the other hand, it would be nice to know if some one actually tried to connect to the router. So we will enable telnet access, we will apply a deny statement and we will log that information.

no access-list 99
access-list 99 deny any log
line vty 0 4
access-class 99 in
login

Router passwords

There are two ways a router's password can be configured – enable password and enable secret. The first one should never be used for a number of reasons – first, even if encrypted, it can be easily decrypted. Next, if no enable secret is set up, and a password for TTY line (console) is set, that console password can be used to get full access to the router. This will work even for telnet sessions on VTY lines.

So, we will configure enable secret and hash it using MD5. The result – it can never be decrypted because using the MD5 hash does not even encrypt the password, it just creates a numeric value with a one-way relationship to the original string. When you type in your enable secret password, the IOS performs the same MD5 hash and compares the results, if they match, you are logged in. As of today, the only way someone could break the enable secret password is to launch a dictionary attack – try every word in a dictionary.

enable secret $^%ER$%##@$
service password-encryption

Warning banner

This is for legal notification requirements:

banner motd #
This is a private system operated by GIAG Enterprises.
Use by unauthorized persons is prohibited.
#

Logging

We will collect information about a variety of events, such as interface status change, changes to the system configuration, access lists matches and intrusion detection system notifications, and send it to the dedicated syslog server 172.16.10.6 on the Management VLAN. Instead of the *log* keyword we will use the *log-input* keyword to enable logging of the IP addresses and port numbers associated with packets matching access list entries.

logging 172.16.10.6

There is no need to filter out packets with broadcast or multicast and loopback IP addresses, the NAT ACL will filter it out.

To prevent a smurf attack, where someone could send ICMP echo requests from a falsified IP address to a directed broadcast address:

no ip directed broadcast

Finally, we will route outgoing traffic to the serial interface, with the exception of the syslog traffic, which should go back to the Management VLAN:

ip route 0.0.0.0 0.0.0.0 Serial0/0
ip route 172.16.10.0 255.255.255.0 Ethernet0/1

Inter VLAN routing.

According to our security policy,

- No VLAN is allowed access to the Management VLAN
- Engineering, Marketing/Sales, Production, HR, Suppliers, and Business Partners VLANs have no access to any other VLANs, except for their own and Firewall VLANs.

IP ranges are as follows:

- Firewall VLAN: 172.16.1.0/24
- Management VLAN: 172.16.10.0/24
- Engineering VLAN: 172.16.20.0/24
- Marketing/Sales VLAN: 172.16.30.0/24
- Production VLAN: 172.16.40.0/24
- HR VLAN: 172.16.50.0/24
- Business Partners VLAN: 172.16.60.0/24
- Suppliers VLAN: 172.16.70.0/24

Here is how it looks from inside the MSFC session:

```
interface Vlan1
 description Firewall VLAN
 ip address 172.16.1.1 255.255.255.0
 ip access-group 101 in
!
interface Vlan10
 description Management VLAN
 ip address 172.16.10.1 255.255.255.0
 ip access-group 110 in
!
interface Vlan20
 description Engineering VLAN
```

```
 ip address 172.16.20.1 255.255.255.0
 ip access-group 120 in
!
interface Vlan30
 description Marketing/Sales VLAN
 ip address 172.16.30.1 255.255.255.0
 ip access-group 130 in
!
interface Vlan40
 description Production VLAN
 ip address 172.16.40.1 255.255.255.0
 ip access-group 140 in
!
interface Vlan50
 description HR VLAN
 ip address 172.16.50.1 255.255.255.0
 ip access-group 150 in
!
interface Vlan60
 description Business Partners VLAN
 ip address 172.16.60.1 255.255.255.0
 ip access-group 160 in
!
interface Vlan70
 description  Suppliers VLAN
 ip address 172.16.70.1 255.255.255.0
 ip access-group 170 in
!
# Firewall VLAN policy
access-list 101 permit ip 172.16.1.0 0.0.0.255 any
access-list 101 deny   ip any any log-input
!
# Management VLAN policy
access-list 110 permit ip 172.16.10.0 0.0.0.255 any
access-list 110 deny   ip any any log-input
!
# Engineering VLAN policy
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.30.0 0.0.0.255
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.40.0 0.0.0.255
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.50.0 0.0.0.255
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 120 deny   ip 172.16.20.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 120 permit ip 172.16.20.0 0.0.0.255 any
access-list 120 deny   ip any any log-input
!
```

```
# Marketing/Sales VLAN policy
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.20.0 0.0.0.255
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.40.0 0.0.0.255
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.50.0 0.0.0.255
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 130 deny   ip 172.16.30.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 130 permit ip 172.16.30.0 0.0.0.255 any
access-list 130 deny   ip any any log-input
!
# Production VLAN policy
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.30.0 0.0.0.255
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.20.0 0.0.0.255
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.50.0 0.0.0.255
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 140 deny   ip 172.16.40.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 140 permit ip 172.16.40.0 0.0.0.255 any
access-list 140 deny   ip any any log-input
!
# HR VLAN policy
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.30.0 0.0.0.255
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.40.0 0.0.0.255
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.20.0 0.0.0.255
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 150 deny   ip 172.16.50.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 150 permit ip 172.16.50.0 0.0.0.255 any
access-list 150 deny   ip any any log-input
!
# Business Partners VLAN policy
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.1.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.20.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.30.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.40.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.50.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 160 deny   ip 172.16.60.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 160 permit ip 172.16.60.0 0.0.0.255 any
access-list 160 deny   ip any any log-input
!
# Suppliers VLAN policy
access-list 170 deny   ip 172.16.70.0 0.0.0.255 172.16.1.0 0.0.0.255
access-list 170 deny   ip 172.16.70.0 0.0.0.255 172.16.10.0 0.0.0.255
access-list 170 deny   ip 172.16.70.0 0.0.0.255 172.16.20.0 0.0.0.255
```

access-list 170 deny ip 172.16.70.0 0.0.0.255 172.16.30.0 0.0.0.255
access-list 170 deny ip 172.16.70.0 0.0.0.255 172.16.40.0 0.0.0.255
access-list 170 deny ip 172.16.70.0 0.0.0.255 172.16.50.0 0.0.0.255
access-list 170 deny ip 172.16.70.0 0.0.0.255 172.16.60.0 0.0.0.255
access-list 170 deny ip 172.16.70.0 0.0.0.255 172.16.70.0 0.0.0.255
access-list 170 permit ip 172.16.70.0 0.0.0.255 any
access-list 170 deny ip any any log-input
!

Here is what all that means – to save space I will review the Suppliers VLAN security policy implementation only:

Access-list 170 is applied inbound to the virtual interface vlan70. First statement denies traffic initiated on the Suppliers VLAN (172.16.70.0/24), going to the Firewall VLAN (172.16.1.0/24) as we need to deny access to the Internet from the Suppliers VLAN.

Next statement denies traffic initiated on Suppliers VLAN (172.16.70.0/24), going to the Engineering VLAN (172.16.10.0/24). And so on.

Two last statements are needed to allow return traffic going back to the Management VLAN and to log any other access list violations; for instance, it will detect and log IP packet spoofing activity on the Suppliers VLAN.

NAT, IOS Firewall Feature Set and IOS IDS configuration

As stated in our Internet access policy, any traffic originated from inside the network, should be allowed to exit, and no traffic originated from the outside world should be allowed in, except for returning traffic, and SSL, HTTP traffic destinated for WWW server 10.0.0.10, ftp traffic destinated for FTP server 10.0.0.11, and SMTP traffic destinated for Exchange 10.0.0.12.

To accomplish that, we will be using Cisco Context Based Access Control feature.

First, we need to enable NAT to translate our internal IP range into something routable.

int e0/1
ip nat inside
!
int s0/1
ip nat outside
!
ip access-list 12 permit 10.0.0.0 0.0.0.255

In this configuration, we will inspect generic TCP and UDP traffic, entering the firewall's external interface. In addition, SMTP and FTP traffic will be examined at the application level, proving another layer of protection, for instance, we will be able to detect and defend from SMTP attacks on our Exchange server, that use packets with illegal commands An illegal command is any command except for the following legal commands: DATA, EHLO, EXPN, HELO, HELP, MAIL, NOOP, QUIT, RCPT, RSET. SAML, SEND, SOML, VRFY. When detected, such packets will be dropped, and the SMTP session will hang and eventually time out.

Here is what needs to be done to configure the CBAC: first, generic TCP and UDP inspection rules need to be created:

Ip inspect name GIAG_Perimiter_Inspection_Rule_1 tcp
Ip inspect name GIAG_Perimiter_Inspection_Rule_1 udp

Next, SMTP and FTP inspections

Ip inspect name GIAG_Perimiter_Inspection_Rule_1 smtp
Ip inspect name GIAG_Perimiter_Inspection_Rule_1 ftp

After that we will create an extended access list 101, denying everything

Access-list 101 deny ip any any

We will apply this access list inbound to the external interface; and the inspection rule needs to applied inbound to the internal interface, so the state table can be maintained for packets originated from the internal network.

To sum up:

ip inspect name GIAG_Perimeter_Inspection_Rule_1 tcp
ip inspect name GIAG_Perimeter_Inspection_Rule_1 udp
ip inspect name GIAG_Perimeter_Inspection_Rule_1 smtp
ip inspect name GIAG_Perimeiter_Inspection_Rule_1 ftp
!
access-list 101 deny ip any any log-input
!
int s0/1

ip access-group 101 in
!
int e0/1
ip inspect GIAG_Perimeter_Inspection_Rule_1 in

At first glance it looks like no traffic is going to enter the internal network because access list 101 denies everything, but in fact, return traffic entering the internal network will be permitted if the packets are part of a valid, existing session for which state information is being maintained – that's exactly what we need to do in order to follow our security policy.

However, we still need to allow some traffic originated from the Internet to enter the DMZ, otherwise no one will be able to connect to our Web, FTP and Exchange boxes.

We will modify the access-list 101 to reflect that:

access-list 101 permit tcp any host XXX.XXX.150.25 eq smtp log-input
access-list 101 permit tcp any host XXX.XXX.150.80 eq www log-input
access-list 101 permit tcp any host XXX.XXX.150.43 eq 443 log-input
access-list 101 permit tcp any host XXX.XXX.150.21 eq ftp log-input
access-list 101 deny ip any any log-input

And static NAT translations will redirect requests to public addresses to their actual addresses:

ip nat inside source static tcp 10.0.0.10 25 XXX.XXX.150.25 25 extendable
ip nat inside source static tcp 10.0.0.12 80 XXX.XXX.150.80 80 extendable
ip nat inside source static tcp 10.0.0.12 443 XXX.XXX.150.80 443 extendable
ip nat inside source static tcp 10.0.0.11 21 XXX.XXX.150.21 21 extendable

In addition, we will define the number of existing half-open sessions and the rate of new un-established sessions allowed before the IOS starts/stops deleting the sessions:

Ip inspect max-incomplete low 500
Ip inspect max-incomplete high 1100
Ip inspect one-minute low 500
Ip inspect one-minute high 1100

IDS configuration

Cisco IOS NIDS is an in-line intrusion sensor that scans packets and sessions to match the IDS signatures.

First we need to initialize IOS IDS, set the threshold beyond which spamming in e-mail messages is suspected, in this case it's 45, and send alarm and attack notifications to our syslog server:

ip audit smtp spam 45
ip audit notify log

Next we need to create audit rules and set the default actions for info and attack signatures:

ip audit info action alarm
ip audit attack action alarm drop reset
ip audit name GIAG_NIDS_RULE_1 info action alarm
ip audit name GIAG_NIDS_RULE_1 attack action alarm drop reset

And finally, we will apply the *GIAG_NIDS_RULE_1* rule in the in direction of the serial interface. This will cause the packets passing through the serial interface to be audited before the inbound ACLs discard them. This will allow us to be alerted if an attack or port scanning activity is underway even if the router would normally reject the packets.

int s0/1
ip audit GIAG_NIDS_RULE_1 in

Site-to-site VPN to Business Partners:

We have a business partner who is using the same IP numbering scheme for its internal network. The connection is a Frame Relay link, below is what needs to be configured on both access routers (3640 at GIAG and 2611 at the Business Partners' site) to establish an IPSEC tunnel between overlapping networks:

1. GIAG router:

First, we will configure IKE parameters, such as encryption, hash and authentication types, define the shared secret key, and specify the interface where the encryption will happen:

crypto isakmp policy 1
encryption 3des
hash md5
authentication pre-share
crypto isakmp key the_key address 192.15.1.57

Next, IPSEC parameters:

crypto ipsec transform-set GIAG_SET esp-3des esp-md5-hmac
crypto map GIAG_MAP 1 ipsec-isakmp
set peer 192.15.1.57
set transform-set GIAG_SET

We will also need to translate overlapping internal IP addresses, so everything on the 172.16.20.0/24 at GIAG will be translated to 172.19.20.0/24, and 172.16.20.0/24 at our partner's subnet will be translated to 172.18.20.0/24

ip nat inside source static network 172.16.20.0 172.18.20.0 /24 no-alias

This will instruct the router to encrypt the traffic going to the other side of the Frame Relay link:

```
interface Serial0/0
 ip address 192.15.1.58 255.255.255.0
 ip nat outside
 crypto map GIAG_MAP
!
match address 135
```

Where access-list 135 allows session establishment between GIAG and our partner's translated addresses:

access-list 135 permit ip 172.18.20.0 0.0.0.255 172.19.20.0 0.0.0.255

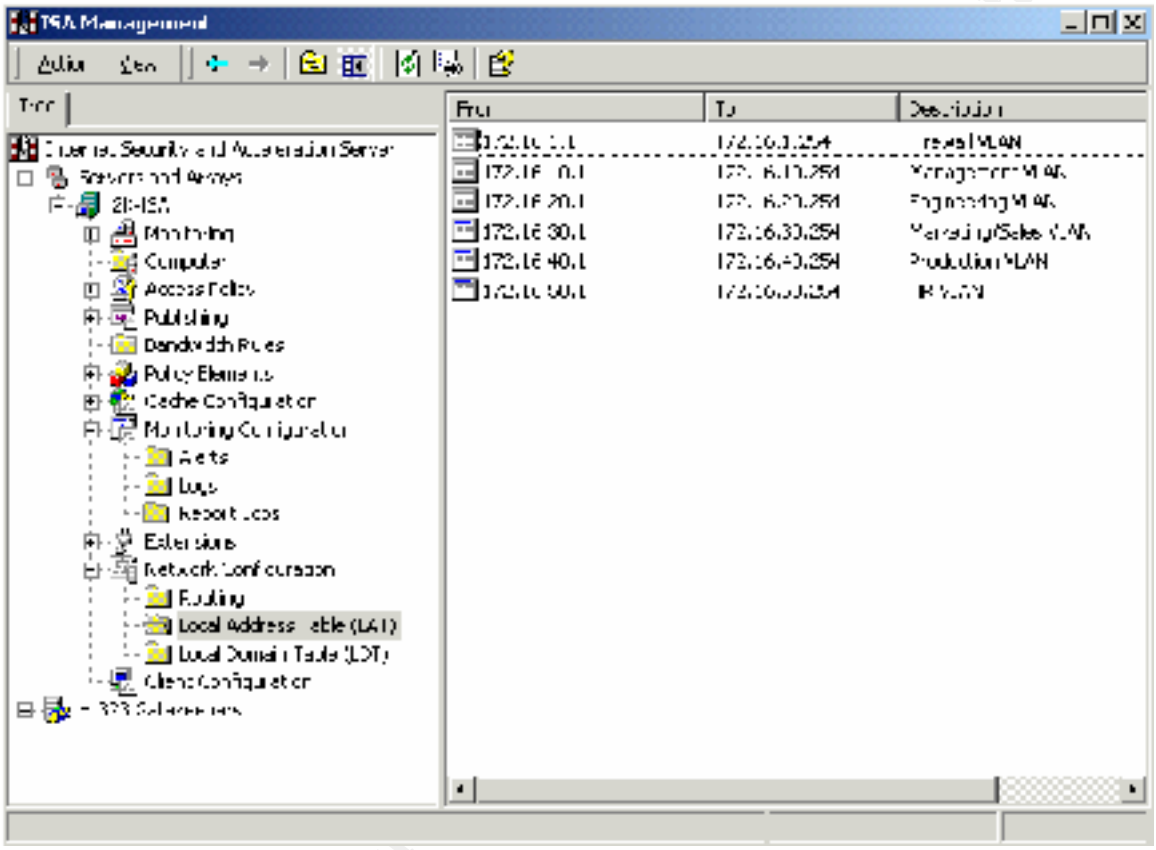
2. The other router's (Cisco 2611) configuration is as follows:

```
crypto isakmp policy 1
 encryption 3des
 hash md5
 authentication pre-share
crypto isakmp key the_key address 192.15.1.58
!
crypto ipsec transform-set GIAG_SET esp-3des esp-md5-hmac
crypto map GIAG_MAP 1 ipsec-isakmp
 set peer 192.15.1.58
 set transform-set GIAG_SET
!
ip nat inside source static network 172.16.20.0 172.19.20.0 /24 no-alias
!
interface Serial0/0
 ip address 192.15.1.57 255.255.255.0
 ip nat outside
 crypto map GIAG_MAP
!
match address 135
access-list 135 permit ip 172.19.20.0 0.0.0.255 172.18.20.0 0.0.0.255
```

ISA Server configuration:

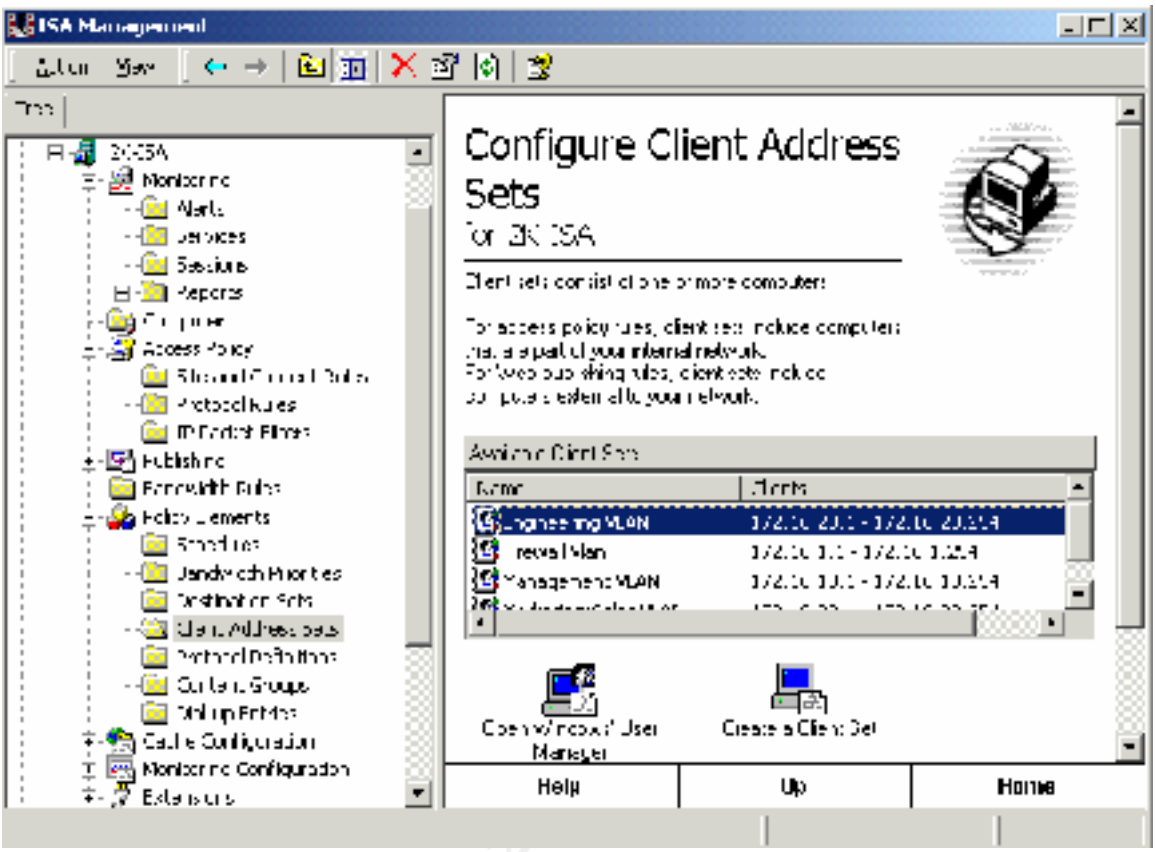
First, we have to create local subnets:

In the ISA Management console, we will create the following entries:

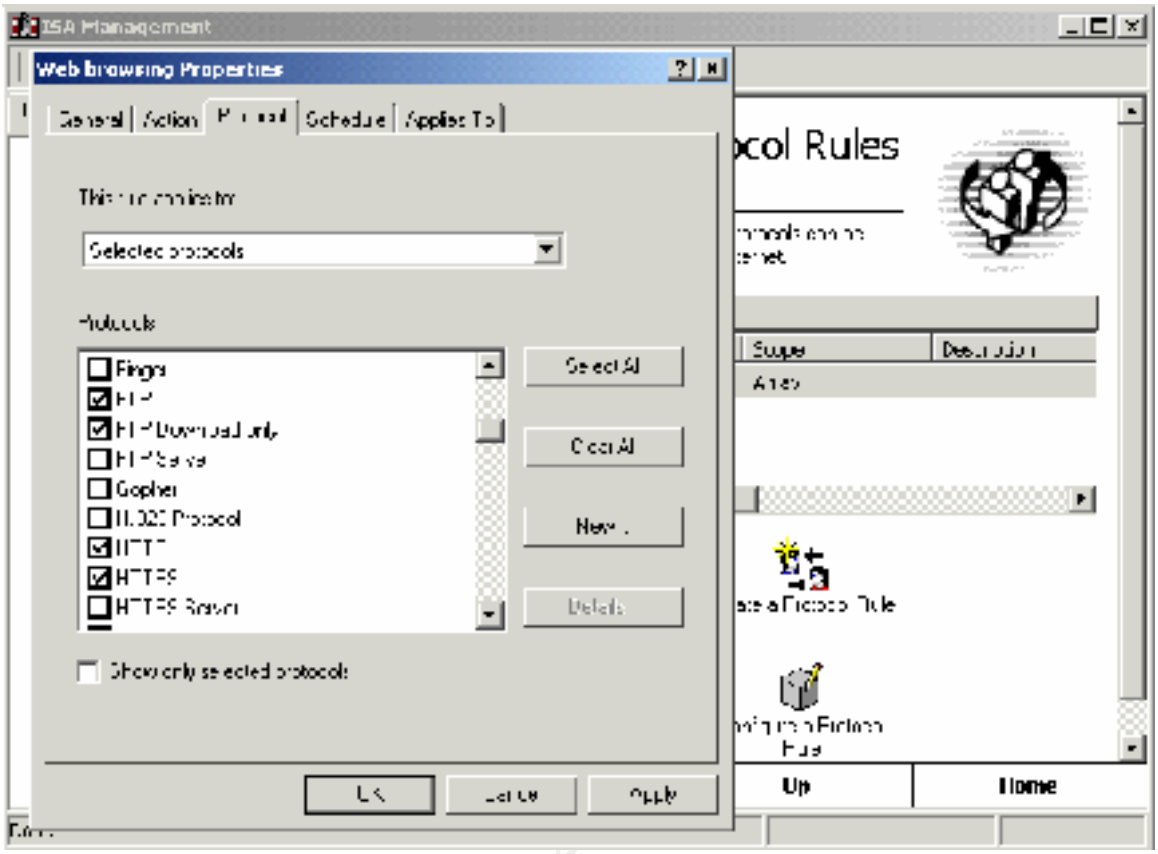


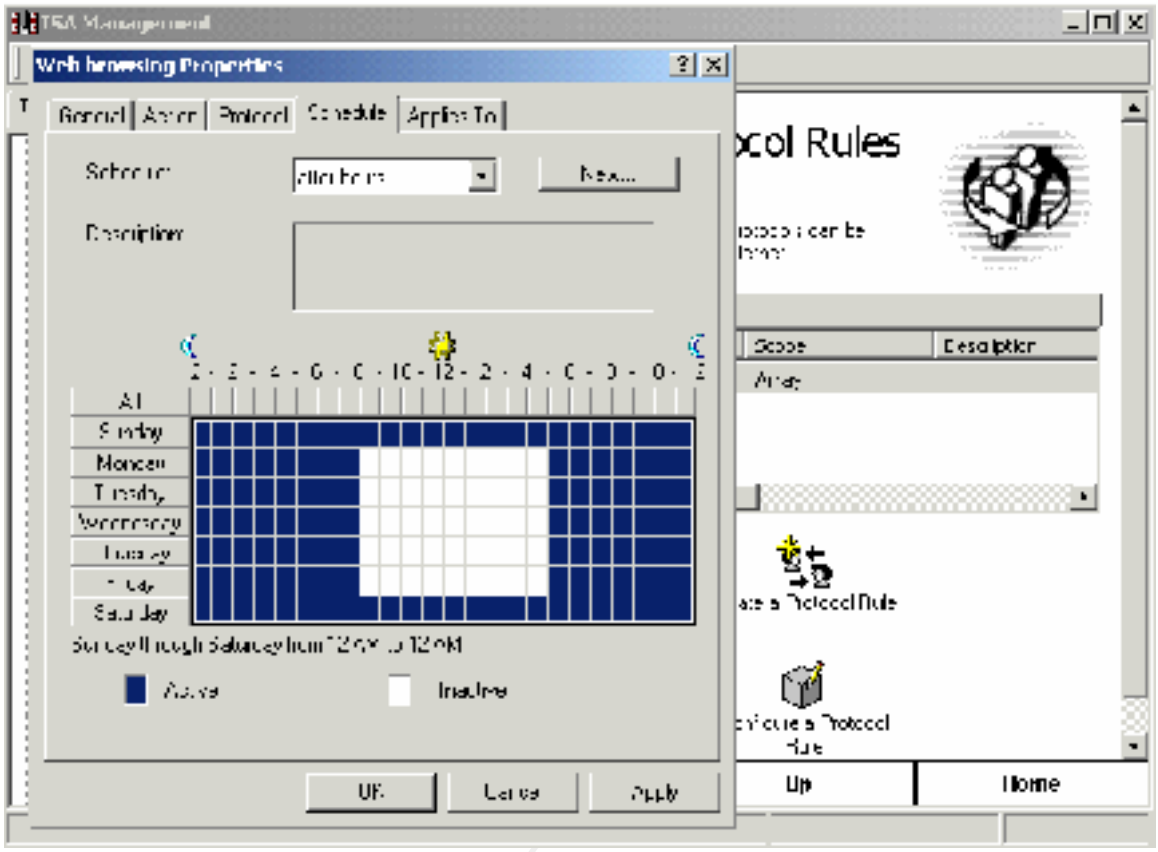
We could actually define LAT as just one entry: 172.16.0.0/16, but we need to exclude the Suppliers and Business Partners VLANs from the translation process. It will be blocked by the 6509's VLAN ACLs anyway, but it won't hurt to filter it out at this point as well.

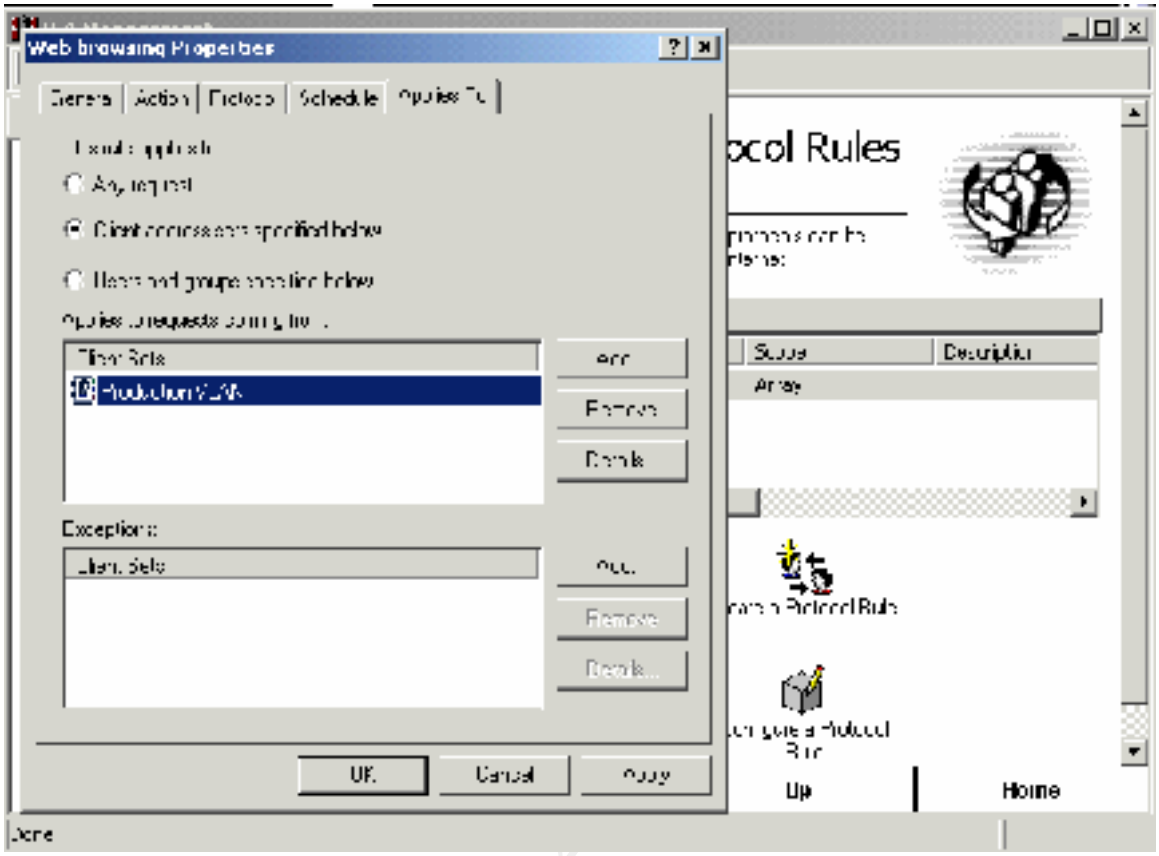
Next, we need to configure client address sets – they will be used to define VLAN policies:



Next, we will define protocol rules for each VLAN. For instance, here is how it looks for the production VLAN. According to our security policy, access to the Internet should be limited to after hours only.
First, we will create the “Web browsing” protocol rule, and define what protocols can be used:







Final step – review the firewall rules. Right out of the box, the ISA server has the following firewall rules enabled:

- DNS filter – **Action:** allow, **Type:** DNS lookup, **Protocol:** UDP, **Direction:** Send Receive, **Local port:** All ports, **Remote port:** 53, **Local computer:** Default external computer, **Remote computer:** Any

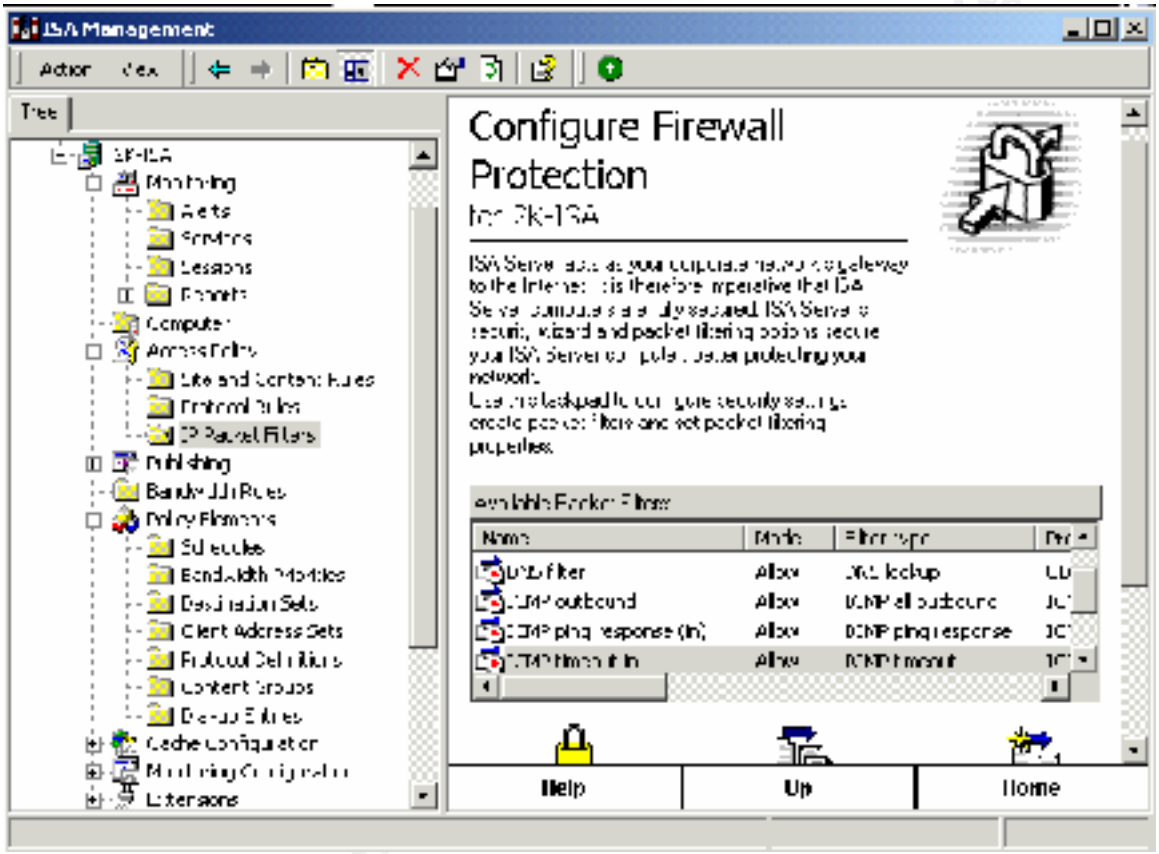
- ICMP outbound – **Action:** allow, **Type:** ICMP all outbound, **Protocol:** ICMP, **Direction:** Outbound, **ICMP Type:** All types, **ICMP code:** all Codes, **Local computer:** Default external computer, **Remote computer:** Any

- ICMP ping response (in) – **Action:** allow, **Type:** ICMP ping response, **Protocol:** ICMP, **Direction:** Inbound, **ICMP Type:** 0, **ICMP code:** all Codes, **Local computer:** Default external computer, **Remote computer:** Any

- ICMP timeout in – **Action:** allow, **Type:** ICMP timeout, **Protocol:** ICMP, **Direction:** Inbound, **ICMP Type:** 11, **ICMP code:** all Codes, **Local computer:** Default external computer, **Remote computer:** Any

- ICMP unreachable in – **Action:** allow, **Type:** ICMP unreachable, **Protocol:** ICMP, **Direction:** Inbound, **ICMP Type:** 0, **ICMP code:** 3, **Local computer:** Default external computer, **Remote computer:** Any

All those rules apply to the ISA server itself, and since we don't use that computer to browse the Internet, we will disable everything. Even if we were using the ISA server to test connectivity with public hosts on the Internet, ping responses would be killed by the IOS CBAC and IDS, in that case the only rule I would enable is the DNS filter in order to be able to browse the Internet from the ISA server.



The firewall and web proxy logs will be checked regularly. In some cases the ISA server can even trace http requests initiated from inside anonymizers, here is one example – the initial connection was made to http://www.anonymizer.com, and from there the actual http request was made to www.privet.com

#Software: Microsoft(R) Internet Security and Acceleration Server 2000
#Version: 1.0
#Date: 2001-07-21 18:18:52
#Fields: c-ip cs-username date time s-computername cs-referred r-host r-ip r-port time-taken cs-bytes sc-bytes cs-protocol s-operation cs-uri s-object-source sc-status
172.16.1.15 anonymous 2001-07-21 18:30:15 2K-ISA - anon.free.anonymizer.com 216.34.244.220 80 6609 452 8552 http GET http://anon.free.anonymizer.com/http://www.privet.com Inet 200

Assignment 3 - Audit Your Security Architecture

You have been assigned to provide technical support for a comprehensive information systems audit for GIAC Enterprises. You are required to audit the Primary Firewall described in Assignments 1 and 2. Your assignment is to:

- *Plan the assessment. Describe the technical approach you recommend to assess your perimeter. Be certain to include considerations such as what shift or day you would do the assessment. Estimate costs and level of effort. Identify risks and considerations.*

The security assessment will consist of two parts – the port scan to see what ports are open, and the actual attack on the perimeter. In our case, two level of defense will be tested separately - the border router, which is also a statefull firewall, and the proxy server (ISA server), which separates the DMZ from our internal network.

The first part will be done after business hours during a workweek. We don't expect the port scan to shut down any hosts and firewalls, the scan is supposed to be filtered out, but just in case it causes a DOS attack if we mis-configured the firewall for instance, we will perform it after hours.

Estimated time for the port scan – 1-2 hour per host.

The second part is the actual attack on the perimeter. Our goal is not to try to penetrate, nothing is 100% secure, if we were going to penetrate the perimeter, we would be able to do that no matter what, but it would not prove anything, so our goal is to check if our perimeter is capable of defending from various well-known attacks. At least our perimeter defense systems should identify and log the attacks, and if possible, drop the sessions.

This is way too much serious than the port scan, and most likely, will cause some hosts to stop responding, so we will perform the test during a weekend.

Estimated time for the port scan – 2-6 hours per host. If a vulnerability is found, and the fix is applied, the attack will be done once again.

All in all, the security assessment will take approximately two days.

Estimated cost will depend on how good our sales guys are.

- *Implement the assessment. Validate that the Primary Firewall is actually implementing the security policy. Be certain to state exactly how you do this, including the tools and commands used. Include screen shots in your report if possible.*

CBAC testing:

If configured properly, the IOS CBAC allows return traffic only from sessions originated from inside the network and denies anything else. In our case, access list 101, applied inbound to the serial interface, denies everything. Below is the output of the "sh access-l 101" command when there is no outbound activity on the internal network, i.e., no one is browsing the Internet:

---------begin output----------
GIAG_3640_A#sh access-l 101
Extended IP access list 101
deny ip any any log-input (20 matches)
GIAG_3640_A#
---------end output----------

Now, when we initiate a web request from a host on the internal network to www.sans.org, the output is completely different, there are temporary openings in the access list 101, that allows return traffic to the originating host:

---------begin output----------
GIAG_3640_A#sh access-l 101
Extended IP access list 101
permit tcp host 167.216.133.33 eq www host XXX.XXX.150.241 eq 4039 (14 matches)
permit tcp host 167.216.133.33 eq www host XXX.XXX.150.241 eq 4037 (60 matches)
permit tcp host 167.216.133.33 eq www host XXX.XXX.150.241 eq 4033 (5 matches)
permit udp host 216.234.97.2 eq domain host XXX.XXX.150.241 eq 3034 (2 matches)
permit udp host 216.234.97.3 eq domain host XXX.XXX.150.241 eq 4030 (2 matches)
deny ip any any log-input (27 matches)
GIAG_3640_A#
---------end output----------

NIDS testing

The network IDS (NIDS) in our case is the Cisco IOS IDS. Packets are being inspected when they enter the serial interface, when packets in a session match a signature, the IDS should send an alarm to the syslog server, drop the packet and reset the TCP connection. Below I will test how the NIDS responds to a few attacks, if it works the way it's supposed to work, we will assume it will work for other known attacks.

Signature 2150, Fragmented ICMP traffic

Triggers when an IP datagram is received with the protocol field in the IP header set to 1 (ICMP) and either the more fragments flag is set to 1 (ICMP) or there is an offset indicated in the offset field. Tested by pinging an internal IP address with payload large than the MTU size, *ping 10.0.0.1 –l 1600*

IDS response:

---------begin output----------
05:15:58: %IDS-4-ICMP_FRAGMENT_SIG: Sig:2150:Fragmented ICMP Traffic - from XXX.XXX.150.241 to10.0.0.1
---------end output----------

Signature 2151 – Large ICMP traffic

Tested by pinging an internal IP address with payload large than 1024 bytes, *ping 10.0.0.1 –l 1025*

IDS response:

---------begin output----------
05:25:49: %IDS-4-ICMP_TOOLARGE_SIG: Sig:2151:Large ICMP Traffic - from XXX.XXX.150.241 to10.0.0.1
---------end output----------

Signature 3151 – FTP SYST Command Attempt.

Triggers when someone tries to execute the FTP SYST command – which usually happens every time you log on to an FTP server.

Tested by initiating an FTP session from the Internet:

---------begin output----------
[root@localhost /root]# ftp XXX.XXX.150.21
Connected to XXX.XXX.150.21
220 ts Microsoft FTP Service (Version 3.0).
Name (XXX.XXX.150.21:root): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 Anonymous user logged in.
Remote system type is Windows_NT
---------end output----------

IDS response:

---------begin output----------

03:18:11: %SEC-6-IPACCESSLOGP: list 101 permitted tcp XXX.XXX.150.1(1650) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(21), 1 packet
03:18:19: %IDS-4-TCP_FTP_SYST_SIG: Sig:3151:FTP SYST Command Attempt - from XXX.XXX.150.1 to 10.0.0.11
---------end output----------

Scanning publicly accessible servers

Scenario: a Linux box in front of the border router, simulating a host on the Internet. We will try to scan our FTP server from the Internet

---------begin output----------
[root@localhost /root]# nmap –P0 XXX.XXX.150.21 –p 1-100
Starting nmap V. 2.53 by fyodor@insecure.org (www.insecure.org/nmap/)
Interesting ports on (XXX.XXX.150.21):
(The 99 ports scanned but not shown below are in state: filtered)

Port	*State*	*Service*
21/tcp	*open*	*ftp*

Nmap run completed – 1 IP address (1 host up) scanned in 46 seconds
---------end output----------

Here is how the CBAC behaved during the scan:

---------begin output----------
02:48:11: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1219) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(76), 1 packet
02:48:11: %SEC-6-IPACCESSLOGS: list 1 denied XXX.XXX.150.241 2 packets
02:48:12: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1257) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(93), 1 packet
02:48:13: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1281) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(10), 1 packet
02:48:14: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1297) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(57), 1 packet
02:48:16: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1312) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(78), 1 packet
02:48:17: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1324) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(18), 1 packet
02:48:18: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1333) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(75), 1 packet
02:48:20: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1348) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(69), 1 packet
02:48:21: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1360) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(45), 1 packet
02:48:22: %SEC-6-IPACCESSLOGP: list 101 permitted tcp XXX.XXX.150.1(1369) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(21), 1 packet

02:48:23: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1386) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(37), 1 packet
……
……
……
0010.a405.2c89) -> XXX.XXX.150.21(97), 1 packet
02:48:42: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1609) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(23), 1 packet
02:48:43: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1621) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(34), 1 packet
02:48:45: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1630) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(88), 1 packet
02:48:46: %SEC-6-IPACCESSLOGP: list 101 denied tcp XXX.XXX.150.1(1645) (Ethernet0/0 0010.a405.2c89) -> XXX.XXX.150.21(63), 1 packet
---------end output----------

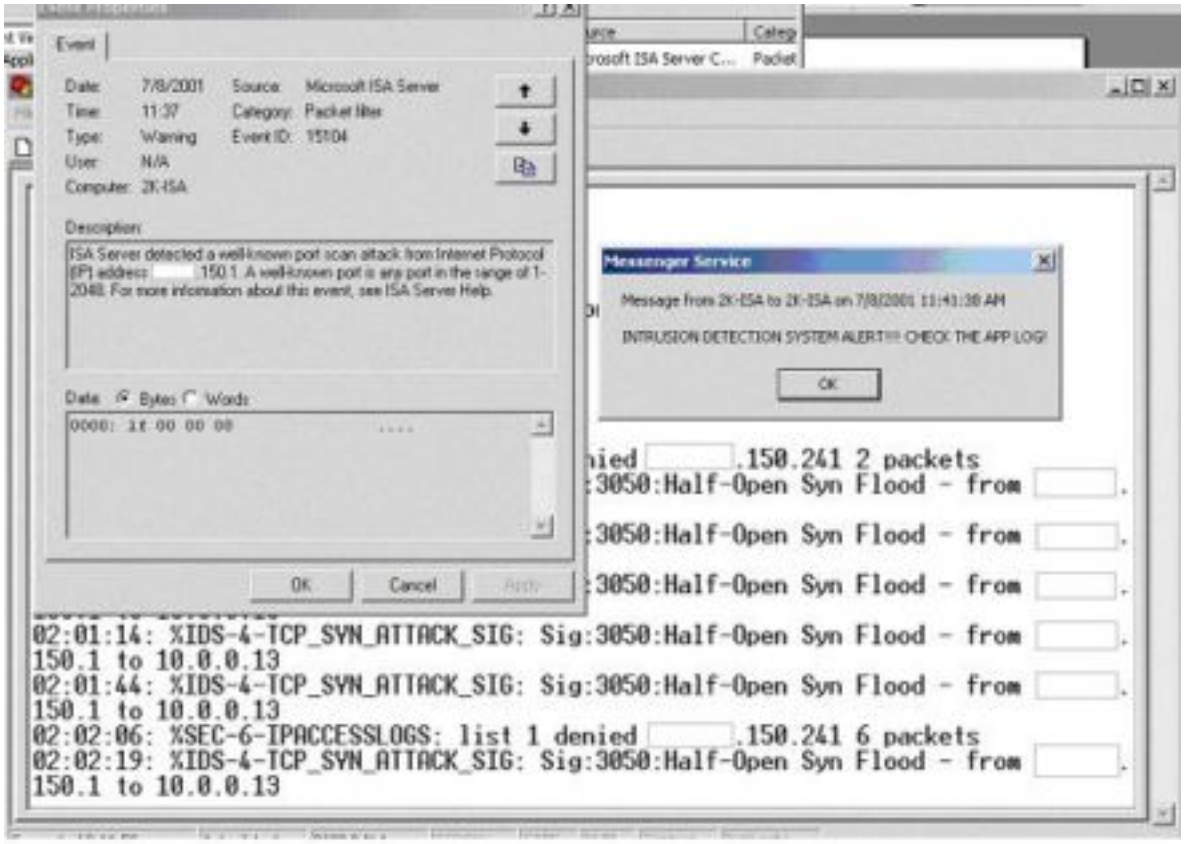
That's exactly how the CBAC should work – it allowed connections to port 21 only and denied connections to any other ports.

Scanning the proxy box

That is the ISA server. Scenario: a Linux box in front of the border router, we are running *[root@localhost /root]# nmap –P0 10.0.0.13 –p 1-65535*

In order to test the HIDS functionality properly, I temporarily disabled the Cisco IOS CBAC by applying the "no access-l 101" command, otherwise everything originated from the outside world would be blocked and the HIDS would never have a chance to see the scan attack.

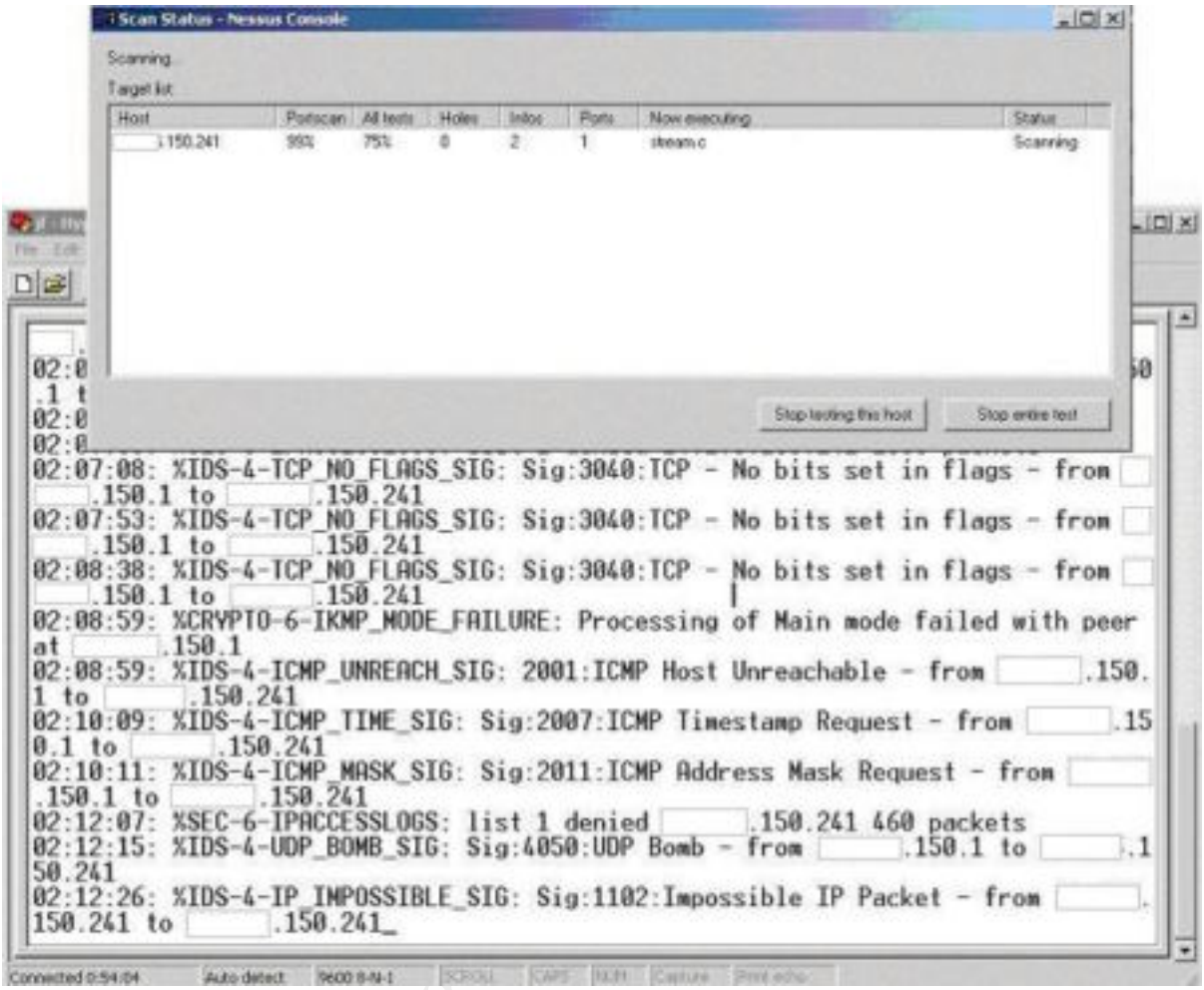
As expected, no ports appear to be opened on the ISA server. In addition, we are getting a lot of alarms from the NIDS (Cisco IOS IDS), as well as from the HIDS (MS ISA server). I configured the ISA server to send a pop-up message every time a possible intrusion is detected; more information about the attack can be found in the application event log. Here is how it looks:



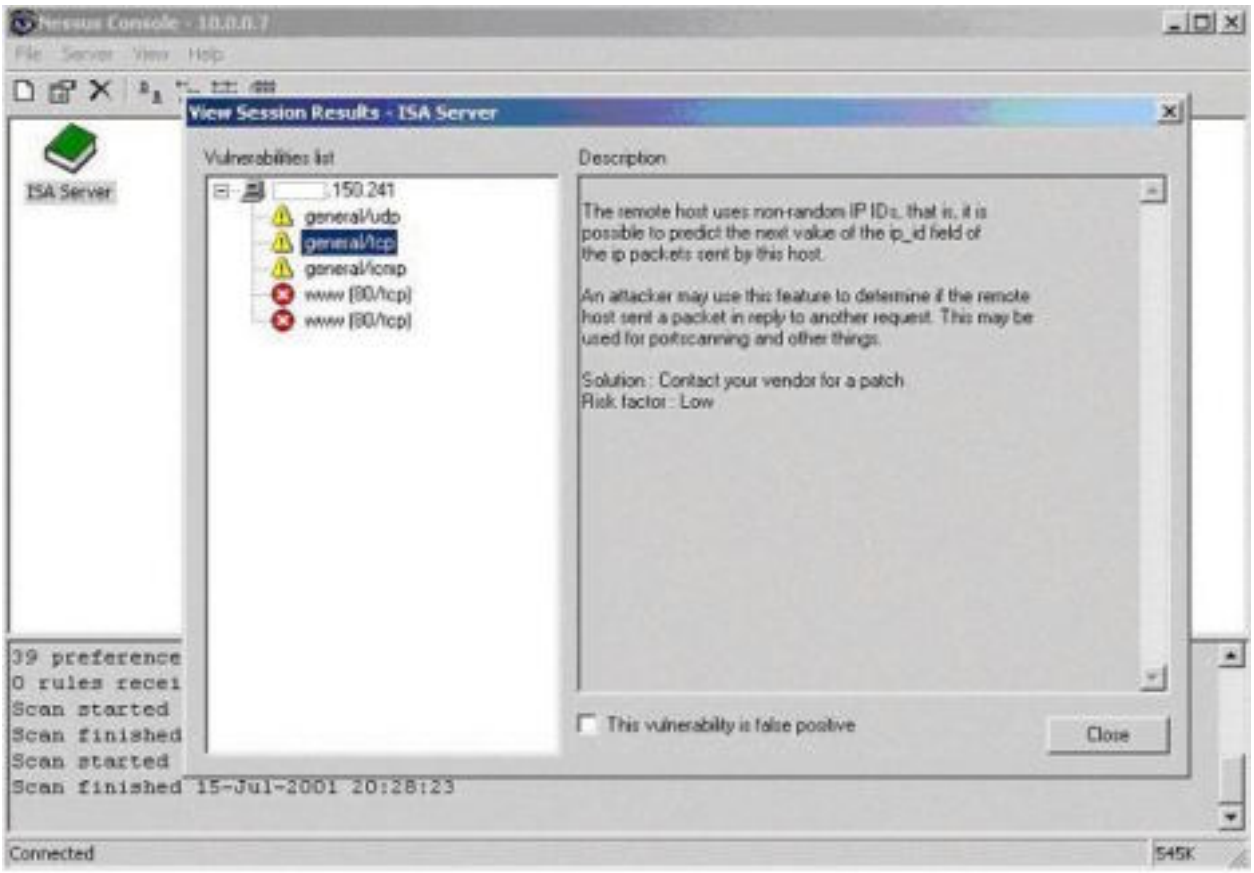
Next part – running a security scanner against the firewall and the proxy box. The software that was used in the security scan was Nessus security scanner. It's a free client-server based scanner running on Unix-like operating systems, offering over 650 security checks for different operating systems. Nessus is made up of two parts – a server and a client.

The test scenario was the following: A Linux box (XXX.XXX.150.1) running nessusd 1.0.8 was installed in front of the border router (XXX.XXX.150.241). A Windows 2000 Server (XXX.XXX.150.2) was running the Nessus console.

First test was to validate that Cisco IOS CBAC and IOS NIDS can handle the attacks. I disabled the access-list 101 to test the IDS functionality; otherwise pretty much everything would be blocked by CBAC. Here is the output of the IOS IDS during the attacks.



No security holes were found during the scan - the IDS successfully identified and reset the suspicious sessions, except for two false vulnerabilities saying that Nessus found MS IIS 4.0 holes on a Cisco router… Next, I enabled CBAC, ran the test again and the result was the same.



The second test scenario looked almost identical, the only difference was that the Linux and W2K boxes were placed behind the border router on the 10.0.0.0/24 network, and the same test was repeated for the ISA server. MS apparently took the security issue seriously enough; no holes were found by Nessus during the scan of the public interface on the ISA Server. Again, as during the nmap scan, the ISA IDS noticed some of the attacks and recorded that to the application log.

- *Conduct a perimeter analysis. Based on your assessment (and referring to data from your assessment), analyze the perimeter defense and make recommendations for improvements or alternate architectures. Diagrams are strongly recommended for this part of the assignment.*

My assessment confirmed that the perimeter is secured adequately, taking into account the budget that was allowed to spend on security technologies.

A few things could be done to improve the existing architecture, for instance, there is absolutely no need for public servers on the DMZ to initiate traffic, and so I would configure another CBAC inspection on the border router to allow the FTP, mail and WWW servers to respond to requests initiated from the Internet, and filter out any other

activity. That would lessen the possibilities that an intruder would get to penetrate the internal network if he/she manages to take over the control of one or more of the DMZ servers.

Also, we have a Network IDS running on the border router and a Host IDS running on the proxy server, but there are no host intrusion detection systems on the public servers on the DNZ. Also, network based IDS rely on signatures that need to be updated regularly, sometimes people forget to do that, and so in addition to the existing NIDS, I would configure a host-based IDS on each server, for instance, Entercept 2.0 from www.entercept.com. According to the product's description, it will prospect hosts from things like buffer overflow exploits, MDAC, Get Admin, it will protect services, processes, registry – everything that's just not possible to protect by ordinary firewalls and intrusion detection systems.

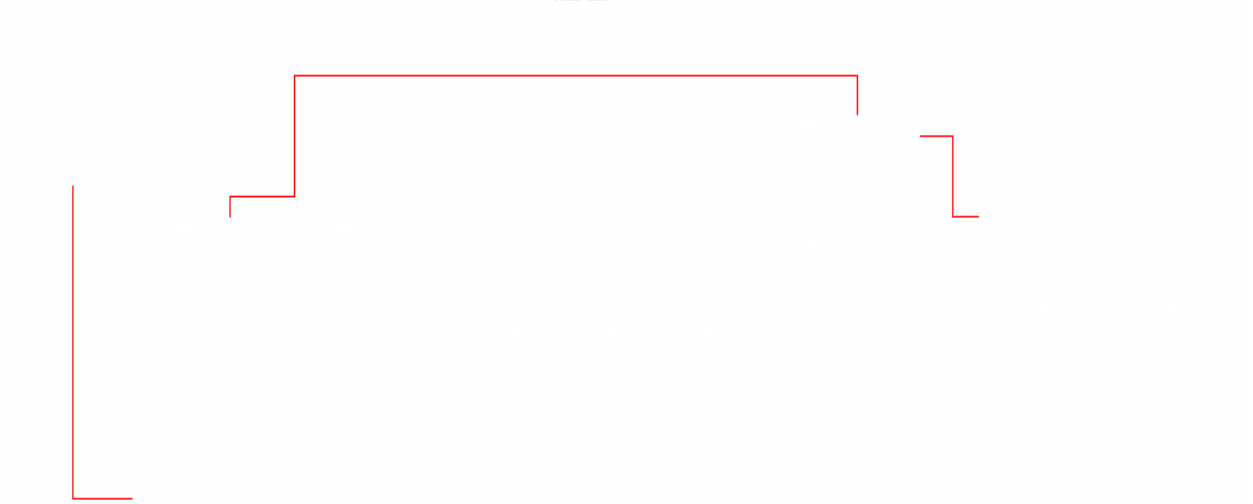
In addition, virus protection software should be installed on each host, especially on the public servers on the DMZ.

Assignment 4 - Design Under Fire

Select a network design from any previously posted GCFW practical (http://www.sans.org/giactc/gcfw.htm) and paste the graphic into your submission. Be certain to list the URL of the practical you are using. Design the following three attacks against the architecture:

- *An attack against the firewall itself. Research vulnerabilities that have been found for the type of firewall chosen for the design. Choose an attack and explain the results of running that attack against the firewall.*
- *A denial of service attack. Subject the design to a theoretical attack from 50 compromised cable modem/DSL systems using TCP SYN, UDP, or ICMP floods. Describe the countermeasures that can be put into place to mitigate the attack that you chose.*
- *An attack plan to compromise an internal system through the perimeter system. Select a target, explain your reasons for choosing that target, and describe the process to compromise the target.*

The practical I am using is http://www.sans.org/y2k/practical/ken_colson_gcfw.doc by Ken Colson.



- *An attack against the firewall itself. Research vulnerabilities that have been found for the type of firewall chosen for the design. Choose an attack and explain the results of running that attack against the firewall.*

The border router that connects the firewall to the Internet is the first device publicly accessible from the Internet, so if I were going to shut down or compromise the firewall I would first try the edge router. It appeared that the router is running "ip http server" and it does not have any content-based access filters, in other words, it allows any traffic originated from the Internet to go trough the router, and it's supposed to be the firewall's job to stop that traffic.

That's bad enough, because if I were a bad guy, I would shut down the edge router in about 5 minutes, which means running a very effective DOS attack against the firewall, no matter how hard it would be to compromise the firewall itself.

Here is how it looks in practice:

All major Cisco IOS versions starting with release 11.3 are affected by the IOS http authorization vulnerability, meaning that if the router is running "ip http server" for Web management and using local database for authentication, it's possible to bypass authentication and execute any command by sending a crafted URL to the routers' http server.

As there is no TACACS+ or RADIUS server on the diagram, and "ip http server" is not disabled, we can try the attack.

First, I would try to point my web browser to the router's public address, let's assume it's 56.67.23.14. If I get a dialog box asking for my user name and password – I know for sure the router is ready to process my http request. I don't know the router's enable or secret passwords so I'll cancel that. Next thing is to point the browser to http:// 56.67.23.14/level/37/exec/- in my case 37 was the magic number. I am in and have full control of the router.

If I were really upset with the company I am attacking, I would do the following:

1. "config t", "enable secret *whatever*"
2. change a few critical parameters, for instance, the IP address of the internal interface – I cannot simply "erase star" – in that case there will be no console password configured and step number 1 would not make much sense. Also I cannot change the IP address of the serial interface at this point because it would kill my connection to the router's http server.
3. "wr"
4. "erase flash"
5. "reload"

The consequences are the following: the router reboots, when it tries to start up, there is no operating system, so it ends up sitting in rommon mode. When someone tries to connect to the router via console, the passwords do not match.

Even if the network manager knows exactly what to do in that situation, and there are copies of the IOS & startup files, it will probably take a few hours to recover the console password, upload the operating system and the configuration file. But it looks like it's not the case because "*ip http server"* was not disabled in the first place. My prediction is that the Internet connection would be down for at least one day.

To defend against this attach, "*ip http server"* should be disabled. If for some strange reason it needs to be enabled, an inbound CBAC filter should be applied to the serial interface, so no connections originated from outside could be accepted by the router's http server.

- *A denial of service attack. Subject the design to a theoretical attack from 50 compromised cable modem/DSL systems using TCP SYN, UDP, or ICMP floods. Describe the countermeasures that can be put into place to mitigate the attack that you chose.*

I selected the mstream DDOS, it's described in detail here:
http://www.cert.org/incident_notes/IN-2000-05.html

There are many places on the Internet where the source files can be downloaded (for instance here http://oliver.efri.hr/~crv/security/bugs/mUNIXes/dos8.html) and compiled.

The mstream tool consists of two pieces - a handler and an agent. Once compiled, the agent in installed on victim machines, in our case, cable modem workstations without any firewall software. The attacker initiates communications with one or more handlers, they, in turn, send commands to compromised hosts.

There are a few commands available for handlers and attackers (compromised hosts running the agent software), as mentioned in
http://oliver.efri.hr/~crv/security/bugs/mUNIXes/dos8.html :

> servers
> - List all currently known agents. The agents may or may not be active. This is the list of all agents that have sent the "newserver" command to the handler at some point in time.
>
> who
> - Shows the currently connected users.
>
> ping
> - Identify remaining active agents. Sends the command "ping" to all known agents and reports to the connected users as each "pong" reply is received.
>
> stream <hostname> <seconds>

- Begin an attack against a single host, for the specified duration. The handler resolves the hostname to an IP address and sends the command mstream/arg1: arg1/arg2" to all agents, where "arg1" is the resolved hosts' IP address twice with a colon between (this simplifies argument parsing in the agent) and "arg2" is the duration in seconds.

mstream <ip1:ip2:ip3:...> <seconds>

- Begin an attack against multiple IP addresses, for the specified duration. The handler sends the command "mstream/arg1/arg2" to all agents, where "arg1" is the list of colon separated IP addresses, and "arg2" is the duration in seconds. Also for simplicity, in this command, there is no host name resolution (i.e., you MUST specify all targets by a properly formed colon separated list of IP addresses)

quit

- Terminates the attacker's connection to the handler. It also reports the termination to the connected users.

Agent Commands

==========

The handler communicates to agents using string based commands in the data portion of UDP packets. These commands are not encrypted (although this, too, can easily be changed.) There are only three agent commands currently. Commands are either a simple string, or a slash ("/") separated command and argument list.

ping

- Replies to IP address that sent this packet with "pong".

stream/IP/seconds

- Starts streaming at IP address for specified duration in seconds.

mstream/IP1[:IP2[:IPN]]/seconds

- Starts streaming at all of the colon separated list of IP addresses for specified duration in seconds.

Even though the agent "in the wild" had two options for accepting DDoS commands, namely "stream" and "mstream" as in the published source, only the mstream command is used in the handler to agent protocol. A simple "stream 192.168.0.100 10" command to the handler sends the powerful command

mstream/192.168.0.100:192.168.0.100/10

to the agent, when in fact a simple "stream/192.168.0.100/10" should be generated. It is not clear why this was done, but it does look like simple

algorithms are used for command parsing, so this might just indicate a "quick and dirty" development process.

To make life more interesting, instead of shutting down and erasing flash on the border router, (remember, I have full control of the router) I could temporarily disable egress filters (so I could use spoofed IP addresses) and redirect the syslog output to a non-existing host during the mstream attack, and return the configuration to the original state after the mstream attack shuts down the Linux firewall (if the border router does not do that first ☺).

- *An attack plan to compromise an internal system through the perimeter system. Select a target, explain your reasons for choosing that target, and describe the process to compromise the target.*

As we have two different types of firewalls in this case, one of which is statefull, it would be quite difficult to penetrate the perimeter, launching a direct network layer attack. I decided to move a few layers up and try an application layer attack.

I will use the QAZ Trojan (http://www.securiteam.com/securitynews/6E00L2000E.html) to get access to the internal network. The scenario would look like this: I would get a few e-mail addresses of the company employees, which can be done very easily searching the newsgroups. Next, I would send an e-mail, disguising as a file called "notepad.exe" - that's what the original version of the QAZ Trojan used.

It does not look like there is anti-virus software installed somewhere on the network, so this e-mail will go through, and chances are, at least one of the recipients will open the attachment. Once it's executed, it renames the original notepad.exe to note.exe and creates a new infected file called notepad.exe.

It also makes some registry changes and activates itself when the systems reboots. When it runs, the QAZ Trojan listens on port 7597 for further commands from the attacker. Using that backdoor channel, it's possible to install and execute different attack tools, for instance, packet sniffers, and gather sensitive information about the internal network such as passwords, directory structure, etc.

If there is unusual activity on port 7597, that means the remote computer send commands and retrieves information from the infected computer. It should never be enabled on the border router. I personally think each Cisco border router should be running CBAC.

References:

Cisco ISP Essentials
http://www.cisco.com/public/cons/isp/documents/IOSEssentialsPDF.zip.

Improving Security on Cisco Routers
http://www.cisco.com/warp/public/707/21.html

A Security Blueprint for Enterprise Networks
http://www.cisco.com/warp/public/cc/so/cuso/epso/sqfr/safe_wp.htm

Extending the Security Blueprint to Small, Midsize, and Remote-User Networks
http://www.cisco.com/warp/public/cc/so/cuso/epso/sqfr/safes_wp.htm

Cisco IOS Firewall Intrusion Detection System
http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t5/iosfw2/ios_ids.htm#22225

Cisco IOS Firewall Feature Set
http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/120newft/120t/120t1/fw3600.htm

Entercept © Security Technologies
http://www.entercept.com/

Microsoft Internet Security & Acceleration Server
http://www.microsoft.com/isaserver/

The Nessus Project
http://www.nessus.org

Nmap scanner
http://www.insecure.org/

RedHat Linux
www.redhat.com

"mstream" Distributed Denial of Service Tool
http://www.cert.org/incident_notes/IN-2000-05.html

"mstream" Distributed Denial of Service Tool
http://oliver.efri.hr/~crv/security/bugs/mUNIXes/dos8.html

QAZ Trojan
http://www.securiteam.com/securitynews/6E00L2000E.html