C2107 Tutorial 6 (Network Security+Access Control)

School of Computing, NUS

March 22, 2021

Background on port number. Some students haven't completed module on Networking and may not familiar with the role of port numbers in server-client connection. In almost all connection, there is a server and a client. The server provides a service and waiting/listening for entities to connect, whereas the client is the one who initiate the connection and request for that service. The service offered by a server can be connected through some predefined port number. E.g. consider the email SMTP server. The server listens to port number 25. If a client wants to connect to the server for SMTP, the client will initiate a connection by sending packets to the server's ip address at port 25. The server responses by sending packets to the client's ip address at another port number (greater than 1023) selected by the client.

Hence, if the client sends a packet to the SMTP server, the destination port of that packet must be 25, but the source port can be any number greater than 1023 selected by the client. Conversely, packets sent by the SMTP has source port 25, and the destination port is the one specified by the client.

The number "25" for SMTP server is an informal standard. A server can choose whatever port number, as long as its clients know about it. Since it is different to announce the port number, servers accessible to the public typically use the standard port number. Potentially, an entity can set up a SMTP server with port number 12332 (just some arbitrary number) and set up its own "private" network of email services. There are methods that attempt to use the port number as a form of secret key– only those who know the port number can connect. Such methods can achieve some form of secrecy but not fool-proof (there are many ways to get that port number).

1. You are the system administrator of a new secondary school and your first task is to design the network and the firewall. You have decided to have 2 to 3 firewalls.

   The machines in the network includes:

   (a) `Lab`. There is a total of around 100 machines in a few labs for students to prepare report, search for materials in the web, etc. There are network printers in the Lab.

   (b) `Teachers`. Every teacher has a PC in the teacher room. The teachers use the PC to enter students' grades, sending/receiving emails, preparing teaching materials, printing exam questions, web-search, etc. There are network printers in the teacher rooms.

   (c) `Web-server`. School's web server.

   (d) `Email-server`. School's SMTP email server.

(e) SQL-server. This is a SQL-server which stores the students database. Some of the information can be accessed through a web-based application hosted in the Web-server (for e.g. allowing student to update their mobile phone number via web-based application). Some information can only be accessed by the teachers.

There is a list of multiple firewall rules in a table. Each rule occupies a row in the table. Ten entries in a row are:

src ip	*dest ip*	*src port*	*dest port*	*direction*	*action*

(a) *Action.* Either Block or Allow.

(b) *src ip.* A set of source ip addresses. It can be specified using the above predefined names in a boolean expression. For e.g. "Email-server". You are free to define other names, e.g. "Internal" to be all machines in the school.

(c) *dest ip.* A set destination ip addresses. Similarly, it can also be a set of predefined names given above.

(d) *source-port, destination-port.* The port number of the source and destination. Note that services use some predefined port numbers. The firewall recognises some fixed port number like HTTP, SMTP, LPR, SQL. (Note: LPR is a network printing protocol listening to port number 515.)

(e) *direction.* This can be IN or OUT. The firewall divides the network into two sectors, say S_1, and S_2. The field indicate which direction the packet is moving, either from S_1 to S_2 or the other way. Your design has to indicate the meaning of IN and OUT.

Same as the example in the lecture note, when a packet arrives, action of the first rule that matches the packet applies. For example, the three rules

src ip	*dest ip*	*src port*	*dest port*	*direction*	*action*
Web-Server	*	HTTP	*	OUT	Allow
*	Web-Server	*	HTTP	IN	Allow
*	*	*	*	*	Block

allows packets sent from the Web-Server to any other ip addresses, and block everything else. (Note that the symbol "*" matches to anything).

Remarks and Requirements.

(a) It is important to prevent cases that exam questions get mistakenly printed in the Lab.

(b) It is important to protect the SQL server.

(c) We know that source ip addresses can be spoofed. The school is worry that some students are running some attack tools that generate spoofed ip-address. Hence they want to block outbound packets from the school that are not having legitimate source ip-addresses.

(d) In this question, we ignore the issue on routing. So, we do not consider the gateway and Network Address Translation. For simplicity, just assume all are "public ip-addresses".

What would be a reasonable partition of the networks and the rules of the firewalls?

Solution

We can place two firewalls, F_1 (front-end/outer firewall) and F_2 (back-end/inner firewall), to separate the public Internet and the school's internal network (`Internal`). In between these two firewalls, we designate the school's DMZ, where the `Web-server`, `Email-server`, `Lab`, and `Lab-printers` are placed. We put `Lab` in the DMZ instead of `Internal` since:

- The `Lab` PCs are considered not to contain any important data; and
- We want to segregate `Lab` and `Teachers` as required.

In `Internal`, we place `Teachers`, `Teacher-printers`, and `SQL-server`.

The network partitioning set-up thus looks like in the following diagram:

`Internal` ← (IN) F_2 (OUT) → DMZ ← (IN) F_1 (OUT) → Internet

The diagram above also indicates the context of `IN` and `OUT` directions on the two employed firewalls F_1 and F_2. We can configure the two firewalls with the rule sets shown in Table 1 and Table 2, respectively.

Remarks:

- Notice that the requirement (c) above, which aims to block outgoing packets with illegitimate source IP addresses (*egress filtering*), is automatically met by the given rule sets.
- Any other Firewall rules can be added as necessary, for examples those needed to allow DNS and HTTPS traffic.

Solution

We can place two firewalls, F_1 (front-end/outer firewall) and F_2 (back-end/inner firewall), to separate the public Internet and the school's internal network (`Internal`). In between these two firewalls, we designate the school's DMZ, where the `Web-server`, `Email-server`, `Lab`, and `Lab-printers` are placed. We put `Lab` in the DMZ instead of `Internal` since:

- The `Lab` PCs are considered not to contain any important data; and
- We want to segregate `Lab` and `Teachers` as required.

In `Internal`, we place `Teachers`, `Teacher-printers`, and `SQL-server`.

The network partitioning set-up thus looks like in the following diagram:

`Internal` $\leftarrow$ (IN) F_2 (OUT) $\rightarrow$ DMZ $\leftarrow$ (IN) F_1 (OUT) $\rightarrow$ Internet

The diagram above also indicates the context of `IN` and `OUT` directions on the two employed firewalls F_1 and F_2. We can configure the two firewalls with the rule sets shown in Table 1 and Table 2, respectively.

Remarks:

- Notice that the requirement (c) above, which aims to block outgoing packets with illegitimate source IP addresses (*egress filtering*), is automatically met by the given rule sets.
- Any other Firewall rules can be added as necessary, for examples those needed to allow DNS and HTTPS traffic.

source IP	*dest IP*	*source port*	*dest port*	*direction*	*action*
Web-server	*	HTTP	*	OUT	Allow
*	Web-server	*	HTTP	IN	Allow
Email-server	*	SMTP	*	OUT	Allow
*	Email-server	*	SMTP	IN	Allow
Lab	*	*	HTTP	OUT	Allow
*	Lab	HTTP	*	IN	Allow
Teachers	*	*	HTTP	OUT	Allow
*	Teachers	HTTP	*	IN	Allow
*	*	*	*	*	Block

Table 1: Firewall rules for the front-end/outer firewall F_1.

source IP	*dest IP*	*source port*	*dest port*	*direction*	*action*
SQL-server	Web-server	SQL	*	OUT	Allow
Web-server	SQL-server	*	SQL	IN	Allow
Teachers	*	*	HTTP	OUT	Allow
*	Teachers	HTTP	*	IN	Allow
Teachers	Email-server	*	SMTP	OUT	Allow
Email-server	Teachers	SMTP	*	IN	Allow
*	*	*	*	*	Block

Table 2: Firewall rules for the back-end/inner firewall F_2.

2. Facebook has many users. There are many objects, including posts and images. Let's use the images as example. How does a user specify who can view the image uploaded by the user? There is some form of "role-based" access control. What is that?

   (As a computing students, even if you prefer not to have a Facebook account, it would be good to create a dummy account so as to know what's happening there.)

Solution

For each post/image, owner can specify

(a) Public

(b) “Friends” except...

(c) Specific friends

(d) Only me

(e) Custom (Include and exclude friends and lists)

(f) “Acquaintances” (Your custom list)

(g) “Photo” (your custom list)

e.g. I (as the owner of this image), specified these two posts as Public, so you should able to see it.

- First cohort of BCOMP Infosec: `https://www.facebook.com/eechien.chang/posts/10216442699642164`
- SOC rooster: `https://www.facebook.com/eechien.chang/posts/10217993940862225`

This one is for “Friends”. So unless you are my “Friends”, you shouldn’t able to see it.

- Having coffee in i4.0 cafe: `https://www.facebook.com/eechien.chang/posts/10221071273633621`

Regarding role-based. The list “Friends”, “Acquaintances”, “custom” and “Photo” are pre-defined list by the Owner. To an owner, when another user B request connection, the owner can specify whether B take up the role of “Friends” or “Acquaintances”.
One could also argue that this is not a truly role-based access control, since typically in role-based access control, the role are defined by the system.

The access control of Facebook posts had being refined over the years and getting more intuitive. Nonetheless, it is still difficult for an owner to fully grasp who has access to the post. (e.g. if a post was specified as “public”, and it contained an image that was specified as “Friends”, what would be the final specification?)