Global Information Assurance Certification Paper

Chris Mahn
GSEC Version 1.2d
21 May 2001

Three Tiered DMZ's

Introduction

Most companies connected to the internet utilize a screened sub-net architecture/DMZ environment to guard against internet threats. The company that wishes to have an internet presence, utilizes a DMZ to segregate their publicly available web servers from their internal network which they deem as a safe zone to be used by their trusted employees.

Part of my job is working as a data security analyst. It always gives me chills when I'm working on a project that puts another web server in our DMZ. Business is business, I can't just raise my hand and say, "No, we're not doing that anymore. The internet community as a whole isn't playing fair any more, so I see no need to take any more risks by placing yet another web server on our DMZ. Have the customer use a phone like they used to." As you can imagine, that wouldn't go over too well. It's not very forward looking. So, how do I get our servers to reside **safely** on our DMZ? After all, my number one job is to make sure our architecture is secure enough to protect our customer's information.

Three Tiered.

For the most part, there are three functionalities that an enterprise's external network provides. It provides an internet presence, SMTP connectivity and DNS resolution. Of these three, the internet presence is what usually creates the most headaches for security.

An internet presence has a web presentation front end, business logic and database services. This is basically a three tiered application. The mistake a lot of companies make is putting their presentation and application servers, along with their email and DNS servers into one DMZ, with their database located on their internal network. This has the same effect as throwing all available keys to your front door, under the doormat. You now have externally initiated traffic crossing your internal network. Once a hacker breaks into a server on the DMZ, they have the permission of the firewall, with certain port and protocol restrictions, to enter. That warm fuzzy feeling you get about your internal network, should now have gone away.

A three tiered DMZ addresses issues surrounding problems with today's single external DMZ architecture, namely security and management, when implemented in conjunction with a standalone DMZ for SMTP and DNS.

This type of DMZ, which directly reflects three tiered applications, with its separate front-end, application and back-end DMZ's, restricts traffic to the company's

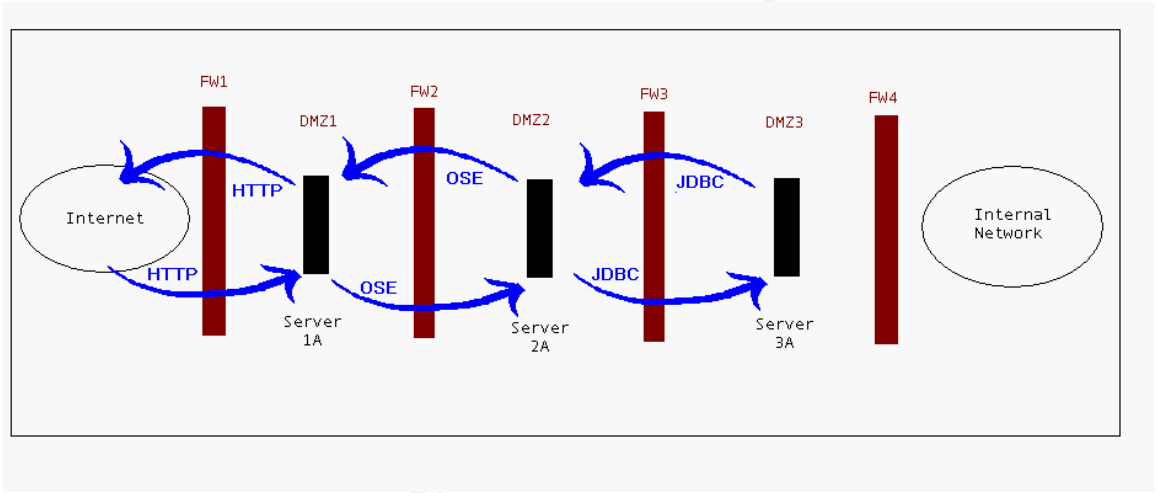
external DMZ, allowing no externally initiated traffic to the company's internal network. So, let me dig in here and elaborate on the above statements.

What does it look like?

You may have asked yourself this already, hopefully I can answer it in a suitable manner. You will recall the brief description of three-tiered applications. The three-tiered DMZ is directly derived from it. Fig 1.a. below is a logical drawing. I want to make note of the fact that physically, this can be implemented any way you choose. From actually having four physical firewalls to using multiple NIC's in one firewall.

This is only the first part. We will build on this design to add the standalone DMZ for SMTP and DNS.

Fig 1.a



The first DMZ, aptly named DMZ1, is the same as most company's external DMZ's. You have the firewall blocking most of the traffic from the internet, except that which you created your DMZ for. Hopefully only port 80 and 443 traffic. We will use the most common type as an example, that being HTTP traffic, which we will allow into our external DMZ to our web server. This is also called the presentation DMZ in a three tiered architecture.

Here the web server will interact with an application server that is located in DMZ2. All traffic will flow through FW2, which will have filters in place to only allow traffic, in this case, from server 1A to server 2A. Here I am using IBM's WebSphere as a web server. It uses a protocol called OSE riding over TCP to communicate with the application server. So, FW2's rule set could be written as:

Accept – IPAddress.of.Server.1A:OSE -----→ IPAddress.of.Server.2A
Deny All;

Server 2A, would then need to get information from its database server located in DMZ3. The traffic has to cross another firewall, FW3. Again, in this example we only have one server in DMZ2, so FW3's rule base could be written as:

Accept – IPAddress.of.Server.2A:JDBC -------→ IPAddress.of.Server.3A
Deny All;

You are, very specifically, narrowing down what traffic is allowed, each step of the way, the further you go into your three-tiered DMZ. If the proper filters are placed at each firewall, your database server is as protected as it had been when it was located on your internal network. Only now you have no externally initiated traffic crossing over that network. Warm fuzzy feeling comes back.

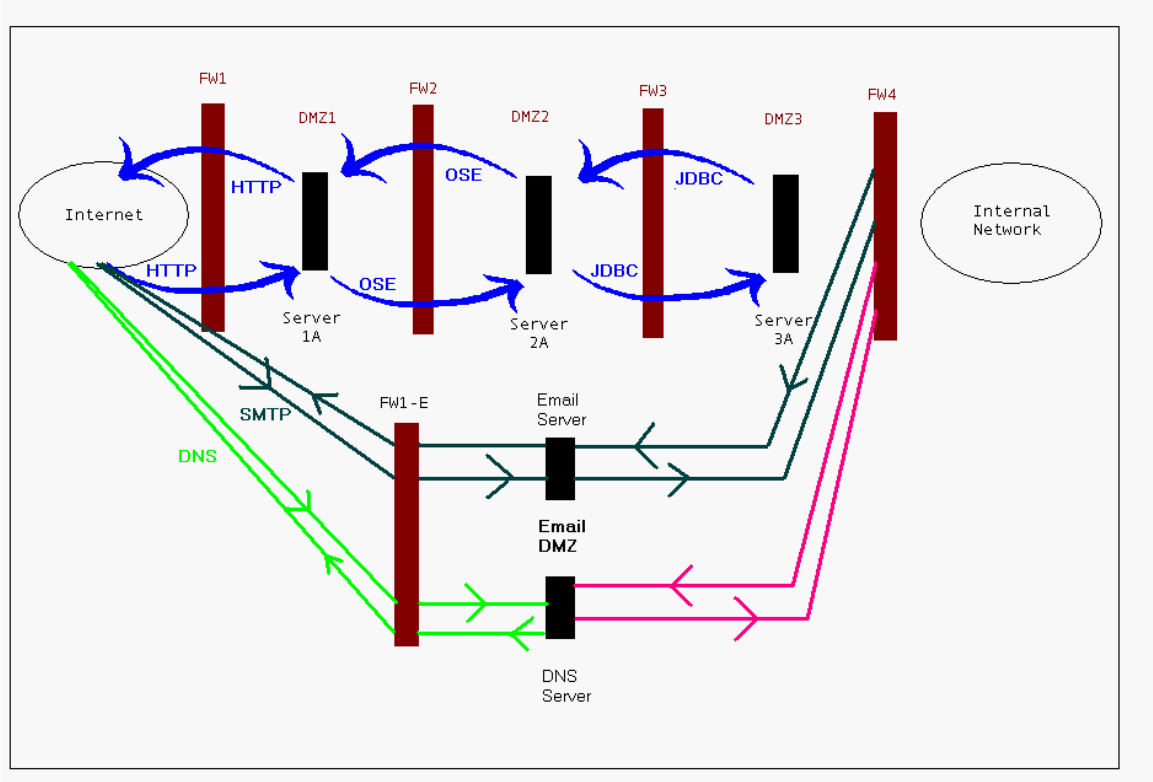
Let me mention a few side notes here. First of all, FW4 would of course not allow any externally initiated traffic whatsoever, into your corporate network. The only traffic coming into your network from this firewall is reply traffic from internally initiated connections to either the DMZ or the internet. Also, DMZ2 should only be allowed to initiate connections to DMZ3, not to DMZ1 or the internet. This same logic applies for DMZ3. There shouldn't be any servers in DMZ3 initiating outbound connections. The only traffic going out should be reply traffic. All this is can be controlled by firewall rules. This will of course stop a trojan infected server, on your DMZ, for instance, from communicating out to the internet. Hopefully with this design, that will not be a problem, but it's always better to be safe then sorry.

In the next section we will talk about certain externally initiated traffic that FW4 allows through, but it will come from an entirely different DMZ.

Adding Email and DNS Services.

In Fig 1.a, what I don't have drawn is the standalone DMZ which would be utilized to add the basic functionality of a corporate network. Below, in Fig 1.b, is the three-tiered DMZ with another DMZ added on for email and DNS. We could call DMZ4 the email DMZ.

Fig 1.b



As you can see, we are still maintaining the integrity of the three-tiered DMZ. Now though, we have added our basic functionality of DNS and Email. FW1-E allows only traffic coming through on port 53 and 25.

In this example, we have one DNS server and one Email server. FW1-E's rule set could look like the following:

Any.IPAddress.on.Internet : 25 -----→ IPAddresss.of.Sendmail.Server

Any.IPAddress.on.Internet: 53 ----→ IPAddress.of.DNS.Server

Deny all;

Email, at this point, would need to be sent to the internal network. This traffic will cross over FW4 into our safe zone. FW4 will only allow SMTP traffic from DMZ4. This is just a high level view. You would narrow this down further to specific destination servers on your internal network.

The DNS server will be the company's external authoritative server. We will have the internal and external zones separated. There is no reason to have internal zones on this DNS server. Therefore you will not have the need for traffic from your external DNS server to your internal DNS server. Also, you would restrict zone transfers from your external DNS server to your ISP's DNS server.

Now we have externally initiated traffic crossing our internal network, but it is a bit more secure. We are only allowing traffic from one DMZ into the internal network, and that DMZ is safely segregated from our publicly available DMZ.

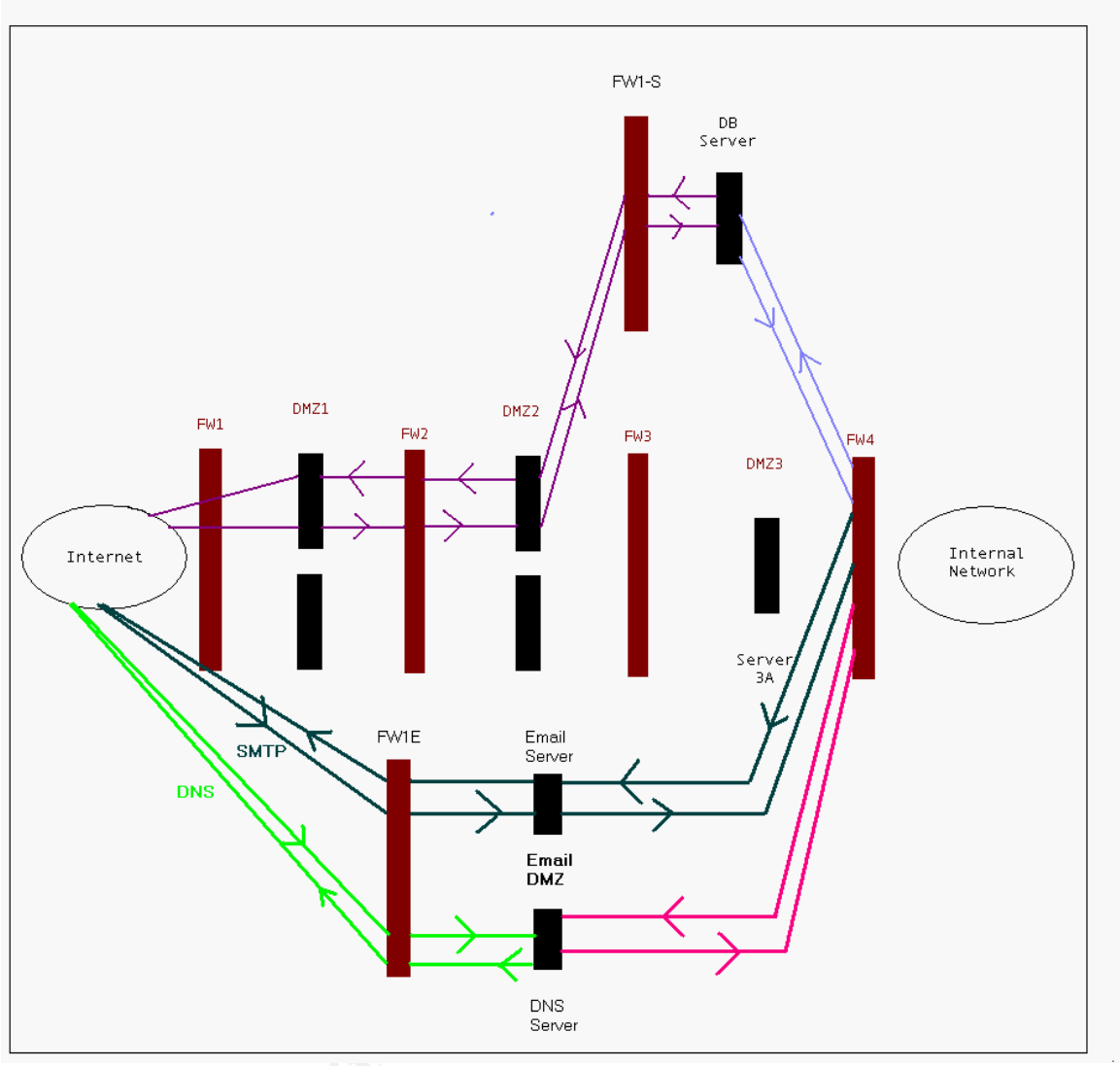
For this to work you would not allow anything else to reside in DMZ4. You could of course extend this idea, and create two separate DMZ's. One for your mail server, and another for your DNS server. I won't bore you with another of my illustrious drawings, but I think you get the idea.

We now have a complete DMZ with our web servers for our internet presence and a standalone DMZ that has our DNS and SMTP services. A three-tiered DMZ can be changed to suit the needs of any company that has an internet presence. This is just the basic concept. Next, we'll flesh it out with a little more functionality.

Co-Accessed Databases.

I am calling them Co-accessed databases for lack of a better word. These are databases that are used by customers and employees. For this we will create a segregated DMZ to place the back-end database in. In fig 1.c below, we have added it as DMZ5. We could have placed the database in DMZ3, but with the open communications we will have to this database from the internal network, it is better to segregate it from DMZ3, which is constantly accessed by the public.

Fig 1.c



The firewall rules are basically the same. We will only allow a server in one DMZ to communicate with a server in the next DMZ and that is restricted to the protocols needed. FW4 will not allow any initiated traffic from DMZ5 to the internal network, only reply traffic. This DMZ could also be used for LDAP servers.

Updating servers in the DMZ.

All the databases that are now on your external DMZ's need some way to be upgraded with O/S and application patches in a secure fashion. The best way to do this is to use a centralized server on the internal network, from which you connect to the database servers and update them. You would punch a hole in FW4 to allow traffic from that specific server, only, on certain ports going to your database servers in the DMZ. Also, you would set a time frame for when this would be accomplished, so the connection is not open all the time. Why leave a door open for a hacker to get through if

you don't have to. This is also how you would treat the maintenance of other servers in your DMZ's.

Additional Considerations.

Network Intrusion Detection Systems.

One of the bigger fallbacks of this type of design is that it requires more network intrusion engines then on a single DMZ. If the company you work for is secure minded enough to look at other ways to make their DMZ tighter, then hopefully this will not be of much concern. If you plan on monitoring all the traffic on the different subnets, this design will definitely force an increase in your intrusion monitoring budget.

Firewall Management.

It looks like this type of design could make it harder to manage your firewalls, but actually, it makes it easier. Once you have the rules in place for your DMZ's, they are mostly static. The only time rules are changed is when another server is added or removed. This leaves a smaller, simpler rule set to maintain for the firewall that restricts traffic coming from the internal network. This is the firewall that has constantly changing rule sets. How many times do you get calls to take such and such a port off of the firewall, only to get another call a little while later to open that port back up.

Added Security.

There are other ideas that you can add to this. One that really sticks out is using reverse proxy. I like the idea of using reverse proxy to a DMZ. There is no direct contact from the internet community to your web servers. All this is handled by your proxy server, which keeps your web servers invisible.

Another added security measure would be to encrypt traffic on your DMZ. There are several methods you could use to do this. You can use IPSEC or open SSH for instance. IPSEC would be harder to implement, but a worthwhile effort considering the advantages of having VPN tunnels from server to server on your DMZ.

Forward Looking Design.

This design lends itself to a forward looking architecture. With this type of DMZ already in place, you can easily expand to accommodate other technology that may be coming down the line. This has to be thought of when considering technology like wireless access, Voice over IP, etc.

Final Thoughts.

In the quest to provide open functionality, security has at times been a hurdle some companies wish to avoid. This will more than likely result in hacked servers or defaced web pages. The three tiered design combines open functionality with security. It

has a forward-looking architecture so a company can expand its internet presence and incorporate new technology as it comes along. In my opinion, that's a pretty good bargain.

One more thought. For the military people our there, shouldn't a DMZ actually be called a Free Fire Zone?

References:

[1] Bellovin, Steven M., "Distributed Firewalls", November 1999. URL: http://www.research.att.com/~smb/papers/distfw.html (10 May 2001).

[2] Curtin, Matt, and Ranum, Marcus J., "Internet Firewalls: Frequently Asked Questions", Version 10, 1 Dec 2000. URL: http://www.interhack.net/pubs/fwfaq/ (10 May 2001).

[3] Nangle, Ken, WebCoach, "Firewalls Guard the Network DMZ", URL: http://www.zdnet.com/sp/infopacks/virus/webcoach.html (10 May 2001).

[4] Sullivan, Aaron, SBC Datacomm Security, "Phase One: The Approach", The Crux of NT Security, 11 Oct 2000, URL: http://www.securityfocus.com/frames/?focus=microsoft&content=/focus/microsoft/nt/crux.html (15 May 2001).

[5] Sullivan, Aaron, SBC Datacomm Security, "Phase Two: Securing The Hosts", The Crux of NT Security, 30 Oct 2000, URL: http://www.securityfocus.com/frames/?focus=microsoft&content=/focus/microsoft/nt/crux2.html (15 May 2001).

[6] Sullivan, Aaron, SBC Datacomm Security, "Phase Three: Controlling and Monitoring Communications", The Crux of NT Security, 22 Nov 2000. URL: http://www.securityfocus.com/frames/?focus=microsoft&content=/focus/microsoft/nt/crux3.html **(15 May 2001).**