Building Codes and Infosec

Standardization For The Win

AverageSecurityGuy

@averagesecguy

stephen@averagesecurityguy.info

patreon.com/averagesecguy

averagesecurityguy.info

What I Do

Network Penetration Testing

Web Application Testing

Development (Python FTW)

Blogging

AppSec Consulting

6110 Hellyer Ave
Suite 100
San Jose, CA 95138
408.224.1110

https://www.appsecconsulting.com

A Few Questions

How far apart are wall studs?

How far apart are floor/ceiling joists?

What is the minimum pitch for a shingled roof?

What is the maximum span of 2 x 8 floor joist?

Reason

Research has shown that these standards provide a strong dwelling structure. Ignoring these standards will get your building project shutdown because the building inspector can be held liable if he/she does not enforce the code.

Advantage

Having a written and enforced building code allows for sound structures, allows practitioners to become proficient quickly, facilitates the transfer of knowledge to the next generation, and allows for easier peer review.

Application to Infosec

Having written and enforced configurations allows for secure infrastructure, allows sysadmins to become knowledgeable of the network quickly, facilitates the transfer of knowledge to other sysadmins, and allows for easier peer review.

Practical Example

If you ask ten framing carpenters to build a wall twelve feet long with a door six feet from the left. Everyone of them will build a very similar, if not identical, wall.

If you ask ten sysadmins to build a web server you will likely get ten completely different builds depending on their backgrounds and training.

It Is Time To Standardize

Keys to Standardization

Write it down: How should a standard server be configured? How should a web server be configured? CIS Benchmarks are a good start.

Automate it: Build an image and use it for all new builds. Scripted installs (learn Bash, PowerShell, Python, or something).

First Steps

Divide all of your assets into broad classes. Workstations, File Servers, Web Servers, Application Servers, etc.

Take the smallest class and document what the standard configuration for that class of machine should be. What OS, software, and configuration? Get your stakeholders involved.

Next Steps

Build a machine that matches your written configuration. What steps can be automated?

Document any changes to the configuration and save any scripts created to automate the build process. This is an iterative process for which virtual machines are well suited.

Start building new machines of this class with your new process.

Continued Progress

Replace existing machines with standardized machines.

Begin standardizing the next largest class of machines.

Don’t forget your routers, firewalls, and switches.

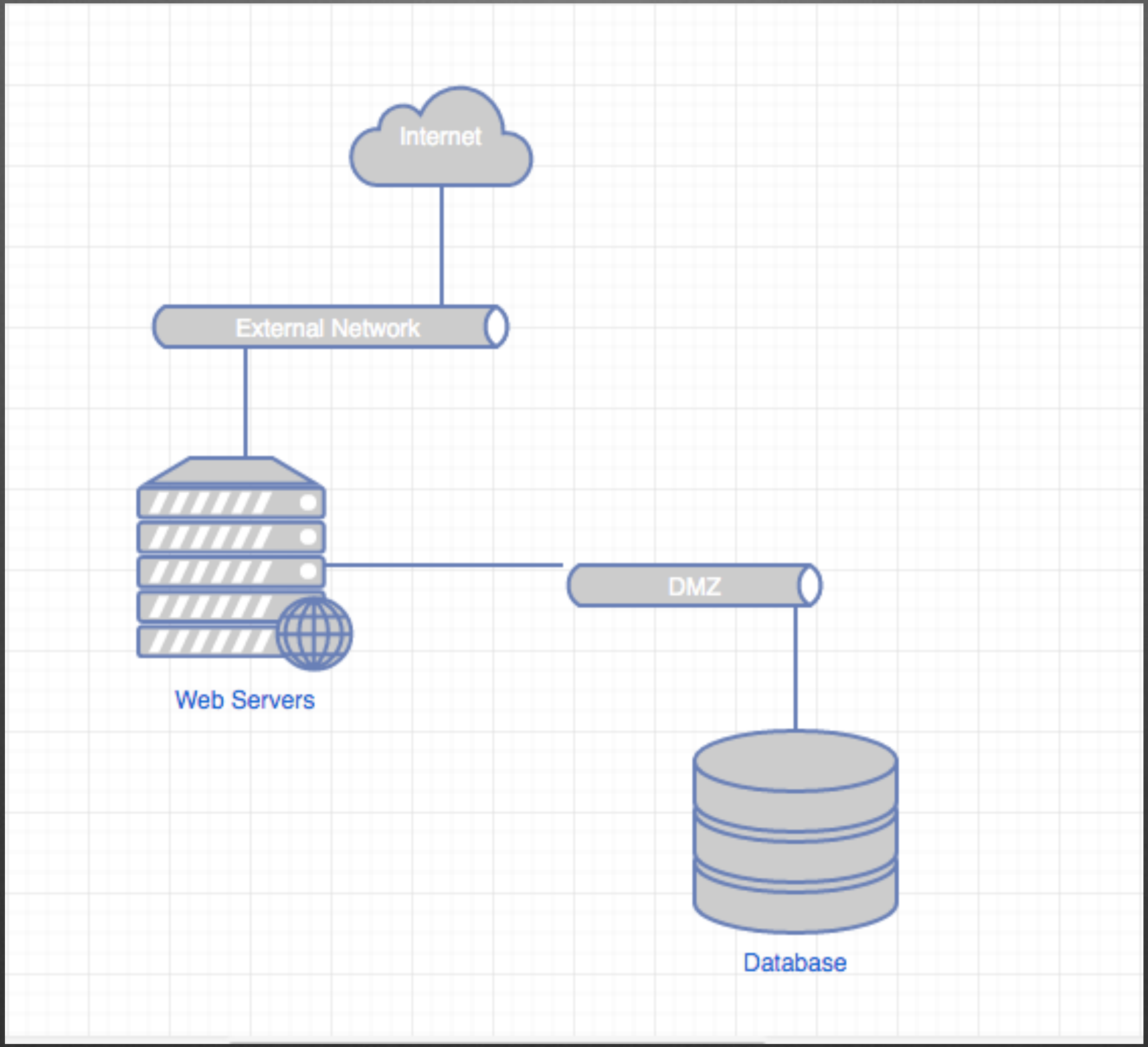
Practice Time

Write a configuration document for a class of servers.

Write a script to implement that configuration.

Build a new server, run the script, and verify the configuration.

Network Diagram



Web Server Requirements

All web servers will run Ubuntu 14.04 LTS

All web servers will run Nginx.

All web servers will be accessed from the DMZ via SSH using an SSH key.

SSH passwords will not be allowed.

Root access to SSH will be disabled.

Web Server Requirements

TCP forwarding and X11 forwarding over SSH will be disabled.

A low privileged user with sudo access will be used (Ubuntu does this by default).

Only ports 80 and 443 will be open on the external network.

Only port 22 will be open on the DMZ.

Script the Requirements

The configuration script is available at https://gist.github.com/averagesecurityguy/752301bf55dc2154d0a8bc4b6376c356#file-web_prep-sh

Demo

Configuration Automation Tools

https://www.chef.io/chef-server/

https://puppetlabs.com/puppet/what-is-puppet

https://www.ansible.com/how-ansible-works

https://docs.saltstack.com/en/latest/

I have not used these tools but many infused folks have. Ask your network.

Thank You

AppSec Consulting
6110 Hellyer Ave
Suite 100
San Jose, CA 95138
408.224.1110

https://www.appsecconsulting.com