Secure Web Services Architecture

A Case Study

Matthew G. Marsh

President, Paktronix Systems LLC

Chief Scientist, NEbraskaCERT

Overview - Section 1

NEbraskaCERT

Web Services

- What is it
- Why is it
- Who cares

Traditional n-Tier Web Services

- MultiTier Arches

Network Security

- Theory
- Reality
- n-Tier
- General
- Tips

Overview - Section 2

- Detailed Analysis of Case Study
  - Systems
    - Multiple Secure Environments
    - Least Privilege
  - Network
    - Protocols
  - Management
    - Updates / Upgrades
- Q & A

S2.1: Systems

Diagram shows standard physical n=3

- Presentation – WWW
- Application – Java
- Data – DB

Each system has common security

Each system has unique security

S2.1.1: Traditional 3-Tier

Presentation (WWW Server)

- Some studies refer to the “Client” tier
- Considering the Client as the “Presentation”
  - JavaScript, HTML, XML

Application (CGI, J2EE, Cobol)

- Not necessarily an independent server
- Best defined by Usage
  - Applications ~= Programs

Data (dBaseIV, SQL, Contacts.TXT)

- Should not imply a DataBase in the operational sense
- Best considered as referential

S2.1.1: Common Security

Each system has common security

- IPTables
- SNORT
- OOB Logging
- Specialized user accounts

S2.1.1: Common Security .2

- IPTables
  - No FORWARD allowed
  - Deny & LOG explicit
    - Sent to OOB
- SNORT
  - Phantom Interface
  - OOB Logging Output (syslog)
- OOB Logging
  - Serial syslog
- Specialized user accounts
  - Apache user
  - Tomcat user
  - Where possible – high level ports binding

S2.1.1: Unique Security

Each system has unique security

- IPTables
- SNORT
- OOB Logging
- Specialized user accounts
- Specialized logging

S2.1.1: Unique Security - .2

IPTables

- Specific INPUT/OUPUT filters

SNORT

- Tuned rules specific to application

OOB Logging

Specialized user accounts

- Lowest privilege possible

Specialized logging

- Per application using OOB where possible

S2.1.2: Multiple Security

- n-Tier Architecture
  - Traditional separation of processing duty.
  - Similar to the concept of an exploded mainframe
- But since this is “ exploded” we can actually obtain access to the points in between
- Even better we can slip in and reside within the middle or back systems
- Each system must adhere to ISN (define a complete PoI structure)

S2.1.3: Least Privilege

Compromise of any specific system MUST NOT compromise any other system

Again – ISN/PoI requirements

Parallel audit trails

OOB management / logging

S2.1.4: Presentation

CIA – Confidentiality, Integrity, Accessibility

- Confidentiality
  - HTTPS / Basic Auth / Certificate structures
- Integrity
  - Protocol Independence / ECC
- Accessibility
  - Availability of system

Apache 2.x system with SSL

- Logging through OOB
  - Consider mod_log_sql
- IPTables only allow connections (INPUT/OUTPUT) to used ports
  - Independent tables filters for each interface

S2.1.4: Application

CIA – Confidentiality, Integrity, Accessibility

- Confidentiality
  - HTTPS / Java Crypto / Certificate structures
- Integrity
  - ECC / JMI
- Accessibility
  - Availability of system

Tomcat 4.x system with SSL

- Logging through OOB
  - JMI / J4L
- IPTables only allow connections (INPUT/OUTPUT) to used ports
  - Independent tables filters for each interface

S2.1.4: Data

CIA – Confidentiality, Integrity, Accessibility

- Confidentiality
  - SSL / mhash / mcrypt / Certificate structures
- Integrity
  - ECC
- Accessibility
  - Availability of system

MySQL 4.0.x system with SSL

- Logging through OOB
- OOB (Serial) management connection
- IPTables only allow connections (INPUT/OUTPUT) to used ports

S2.2: Networks

Connections between each system using different RFC1918 network.

Presentation server had default route outbound only

Application and Data server had no default routes

Full Policy Routing structures

- ip rule limited accesses exiting localhost
- No IPv4 forwarding

S2.3: General Notes

SNMP

- Use IPX where possible
- Use Version 3 with full authPriv and Inform traps
  - Separate passphrases for auth and Priv

Serial Logging

- AKA Out Of Band (OOB) Logging
- Consider two serial connections – one in & one out

Time Synchronisation

- Does not need to be accurate merely precise

Read Only DASD

- Especially useful for static content
- Works well with well behaved programs (Apache)

SSH / SSL

This is The