APPDEV AND SECARCH – BUILDING A BETTER TOMORROW TOGETHER!

BY

CROB

MELANIE MCKEAN

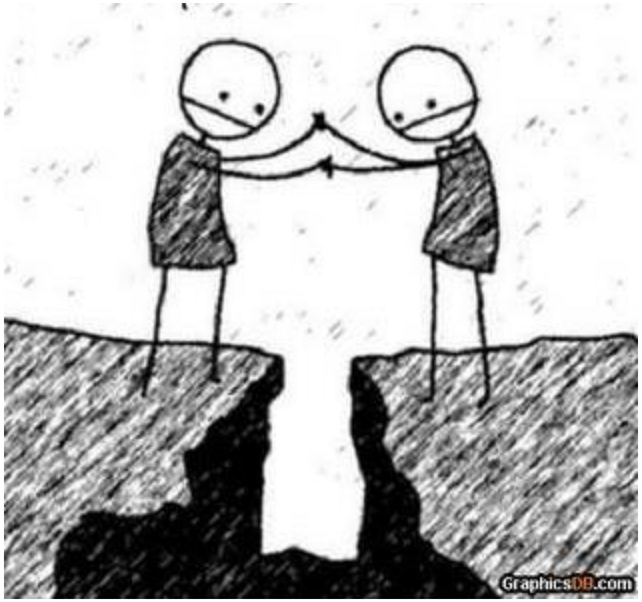
BIOGRAPHIES

Christopher Robinson (aka CRob) is the Information Security Architect for Westfield Group. With over 18 years of Enterprise-class engineering, operational and leadership experience, Chris has worked at several Fortune 500 companies with experience in the Financial, Medical, Legal, and Manufacturing verticals. CRob has been a featured speaker at Gartner's Identity and Access Management Summit, RSA, as well as several Tivoli Pulse and Splunk conferences. CRob is active in the Information Security community. He is the Education Officer for the Cleveland (ISC)2 Chapter.

Melanie McKean is an Application Architect at Westfield Group, with the focus on people, processes, and technologies for distributed development on the JEE platform. She is an integral part of ensuring the promotion and implementation of security throughout the software development lifecycle for Westfield. Her most recent assignment on an Enterprise-focused system update included the secure propagation of identity between applications and services through the implementation of several IBM security tools working in concert with WebSphere Application Server (WAS).

IN THE BEGINNING…

Application Development Says……..



Information Security Says……

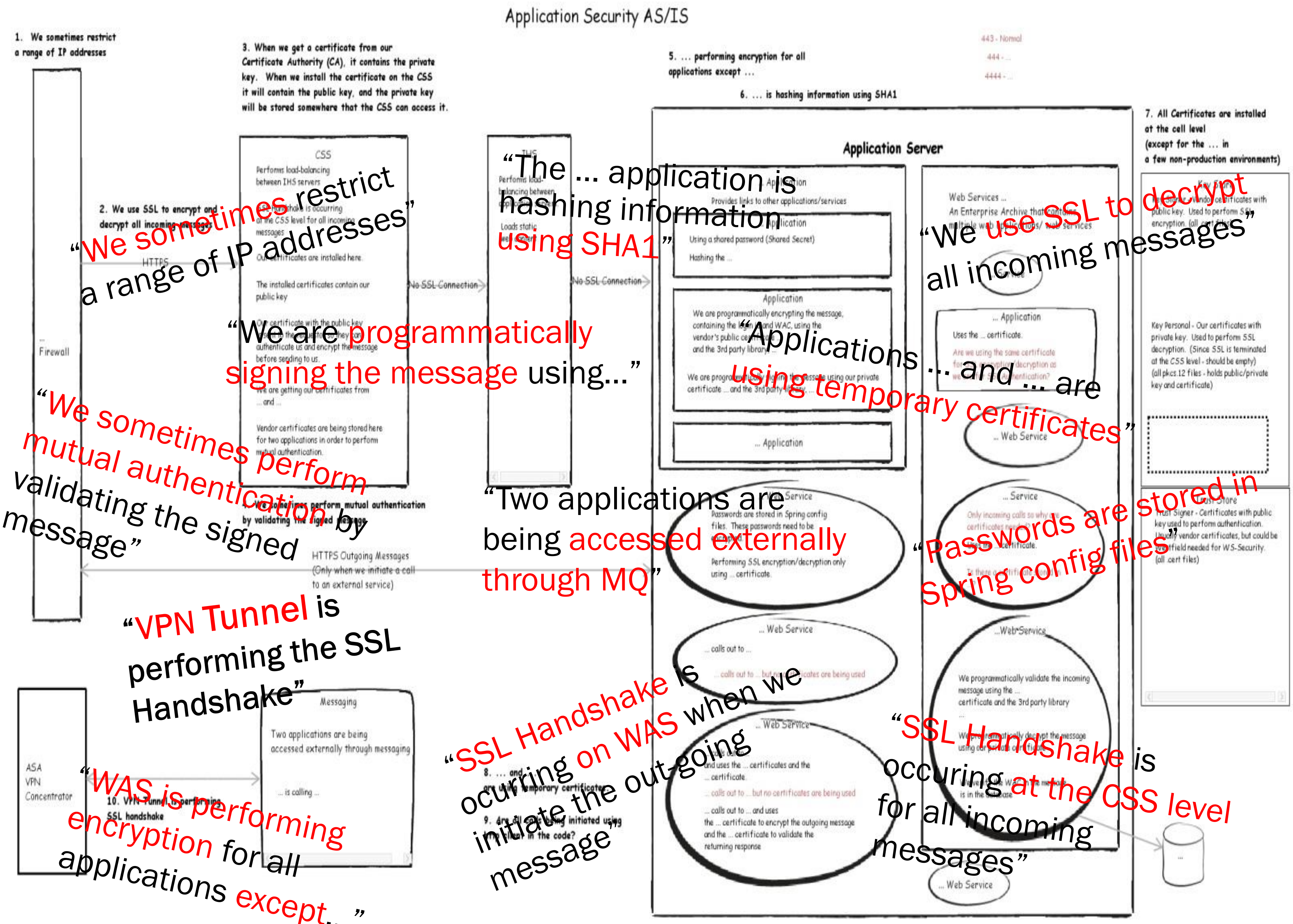
OWASP To 10 Web Application Security Risks

- WebGoat Hands-On Learning
- Cheat Sheets

Kerberos / SPNEGO
LTPA – Lightweight Third-Party Authentication
TAI – Trusted Association Interceptor
Single Sign-On
AccessManager
Digital Certificates
Digital Signing
SSL
PKI
SAML
SAMM
Encryption
Symmetric / Asymmetric

WS-Security
WS-Secure Conversation
WS-Federation
WS-Context
WS-Policy
WS-Authorization
WS-Trust
WS-Privacy

Authentication Service
Authorization Service
Audit Service
Web Service Gateway


Application Security AS/IS
1. We sometimes restrict a range of IP addresses
Firewall
2. We use SSL to encrypt and decrypt all incoming messages
HTTPS
3. When we get a certificate from our Certificate Authority (CA), it contains the private key. When we install the certificate on the CSS it will contain the public key, and the private key will be stored somewhere that the CSS can access it.
CSS
Performs load-balancing between IHS servers
SSL Handshake is occurring at the CSS level for all incoming messages
Our certificates are installed here.
The installed certificates contain our public key
Our certificate with the public key
authenticate us and encrypt the message before sending to us.
We are getting our certificates from … and …
Vendor certificates are being stored here for two applications in order to perform mutual authentication.
4. We sometimes perform mutual authentication by validating the signed message
No SSL Connection
IHS
Performs load-balancing between application servers
Loads static web content
No SSL Connection
HTTPS Outgoing Messages (Only when we initiate a call to an external service)
5. … performing encryption for all applications except …
443 - Normal
444 - …
4444 - …
6. … is hashing information using SHA1
Application Server
… Application
Provides links to other applications/services
… Application
Using a shared password (Shared Secret)
Hashing the …
Application
We are programmatically encrypting the message, containing the login and WAC, using the vendor's public certificate … and the 3rd party library …
We are programmatically signing the message using our private certificate … and the 3rd party library …
… Application
Web Services …
An Enterprise Archive that contains multiple web applications/ web services
… Service
… Application
Uses the … certificate.
Are we using the same certificate for SSL encryption/decryption as we are for authentication?
… Web Service
… Web Service
Passwords are stored in Spring config files. These passwords need to be encrypted.
Performing SSL encryption/decryption only using … certificate.
… Service
Only incoming calls so why are certificates needed?
Uses the … certificate.
Is there a certificate stored in …
… Web Service
… calls out to …
… calls out to … but no certificates are being used
… Web Service
… calls out to … and uses the … certificates and the … certificate
… calls out to … but no certificates are being used
… calls out to … and uses the … certificate to encrypt the outgoing message and the … certificate to validate the returning response
… Web Service
We programmatically validate the incoming message using the … certificate and the 3rd party library
…
We programmatically decrypt the message using our private certificate
We verify the WAC in the message is in the database
… Web Service
…
7. All Certificates are installed at the cell level (except for the … in a few non-production environments)
Key Store
Trust Signer - Vendor certificates with public key. Used to perform SSL encryption. (all cert files)
Key Personal - Our certificates with private key. Used to perform SSL decryption. (Since SSL is terminated at the CSS level - should be empty) (all pkcs12 files - holds public/private key and certificate)
Trust Store
Trust Signer - Certificates with public key used to perform authentication. Usually vendor certificates, but could be … field needed for WS-Security. (all cert files)
ASA VPN Concentrator
10. VPN Tunnel is performing SSL handshake
Messaging
Two applications are being accessed externally through messaging
… is calling …
8. … and … are using temporary certificates
9. Are all calls being initiated using … in the code?

"We sometimes restrict a range of IP addresses"

"The … application is hashing information using SHA1"

"We use SSL to decrypt all incoming messages"

"We are programmatically signing the message using…"

"Applications … and … are using temporary certificates"

"We sometimes perform mutual authentication by validating the signed message"

"Two applications are being accessed externally through MQ"

"Passwords are stored in Spring config files"

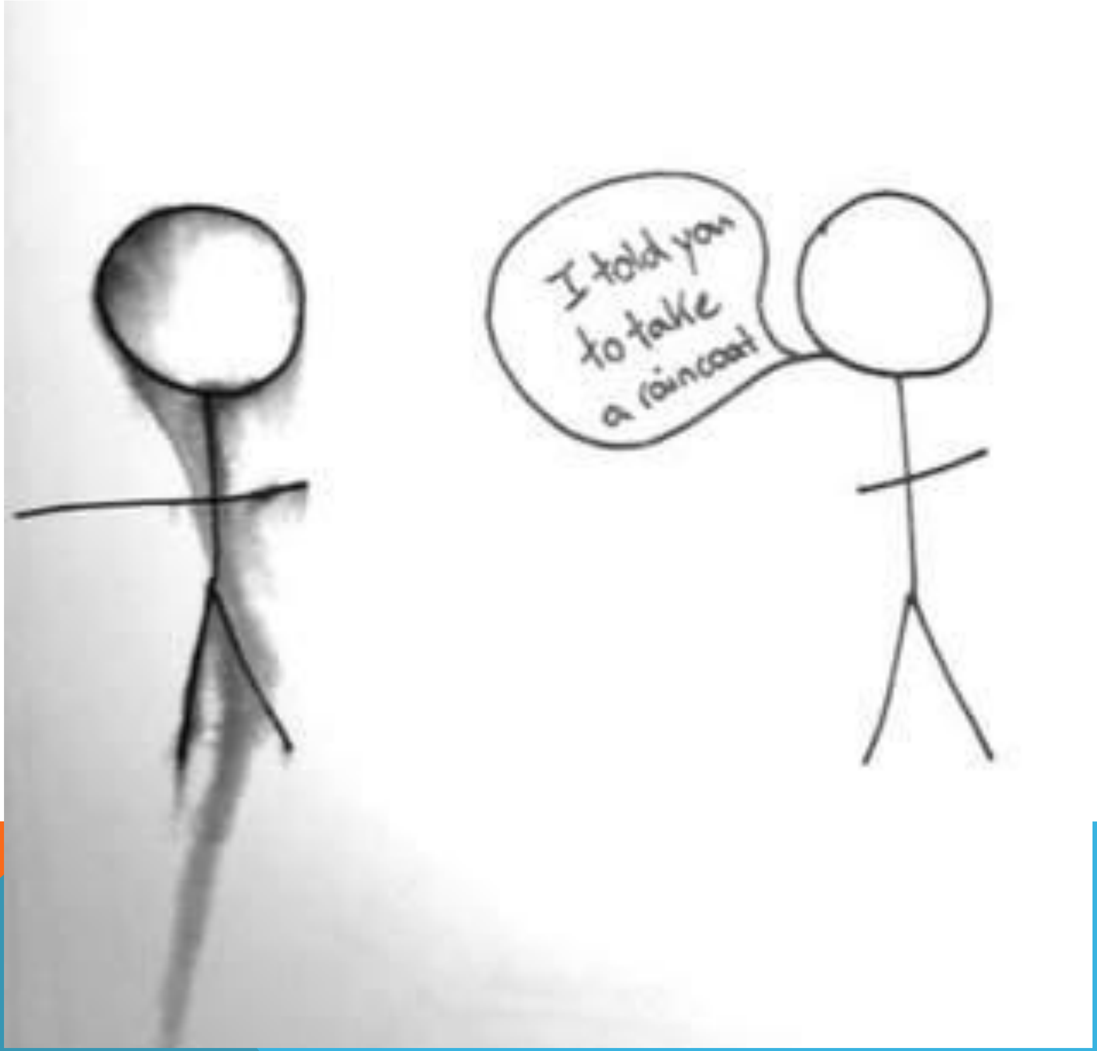
"VPN Tunnel is performing the SSL Handshake"

"SSL Handshake is ocurring on WAS when we initiate the out-going message"

"SSL Handshake is occuring at the CSS level for all incoming messages"

"WAS is performing encryption for all applications except…"

MILESTONE 1….



NEW CLAIMS MANAGEMENT SYSTEM

ARCHITECTURAL AND SECURITY REQUIREMENTS

- Simplified Sign-On
- Propagate user identity between systems
- Federate user identity with vendor systems
- Secure Integrations

SIMPLIFIED SINGLE SIGN-ON

- *Complex business processes require the user to interact directly with multiple systems as part of the normal workflow. Simplified sign on improves user efficiency, which is a part of the ROI on modern systems. XSA simplified sign on allows for a common user (web) sign on (SSO) to multiple systems as well as common session management and coordinated session recovery between systems*

IMPLEMENTATION OF SINGLE SIGN-ON

Proxy Server

- WebSEAL

Passing credential information

- eTai+ to establish JAAS Subject
- LTPA Token to establish JAAS Subject

Authentication

- TDS to Active Directory

Coarse Grained Authorization

- ACLs – TDS

PROPAGATE USER IDENTITY BETWEEN SYSTEMS

- *Enterprise systems are integrated using services and SOA to enable complex business processes and information sharing. In some cases there is a business need to propagate user context between systems as part of the service call since the user has different levels of authorization in those systems. The identity must be propagated at a system level that is transparent to the user.*

IDENTITY PROPAGATION

WS-Security

- SAML 2.0

Security in the Infrastructure

- Industry Standards – JAX-WS
- JAAS Subject

Encryption / Signing

- Local token validation – exception GWCC

Single Point of Token Generation

Policy Sets and Bindings

- App Configs
- XML files

FEDERATE USER IDENTITY WITH VENDOR SYSTEMS

- *Our business processes demand secure, real-time integration of our internal systems with vendor and cloud-based systems. Identity federation enables sharing and managing identity information with vendors to allow for simplified sign on and propagating user identity between internal and external systems.*

USER IDENTITY FEDERATION

Web Applications

- Required additional login
- Special Case using WebSEAL

Web Services

- Web Services Gateway
- Outbound
- Inbound Integration with FIM

SFTP

- FTP Server
- Inbound Integration with FIM

Future Federated User Identity Solutions

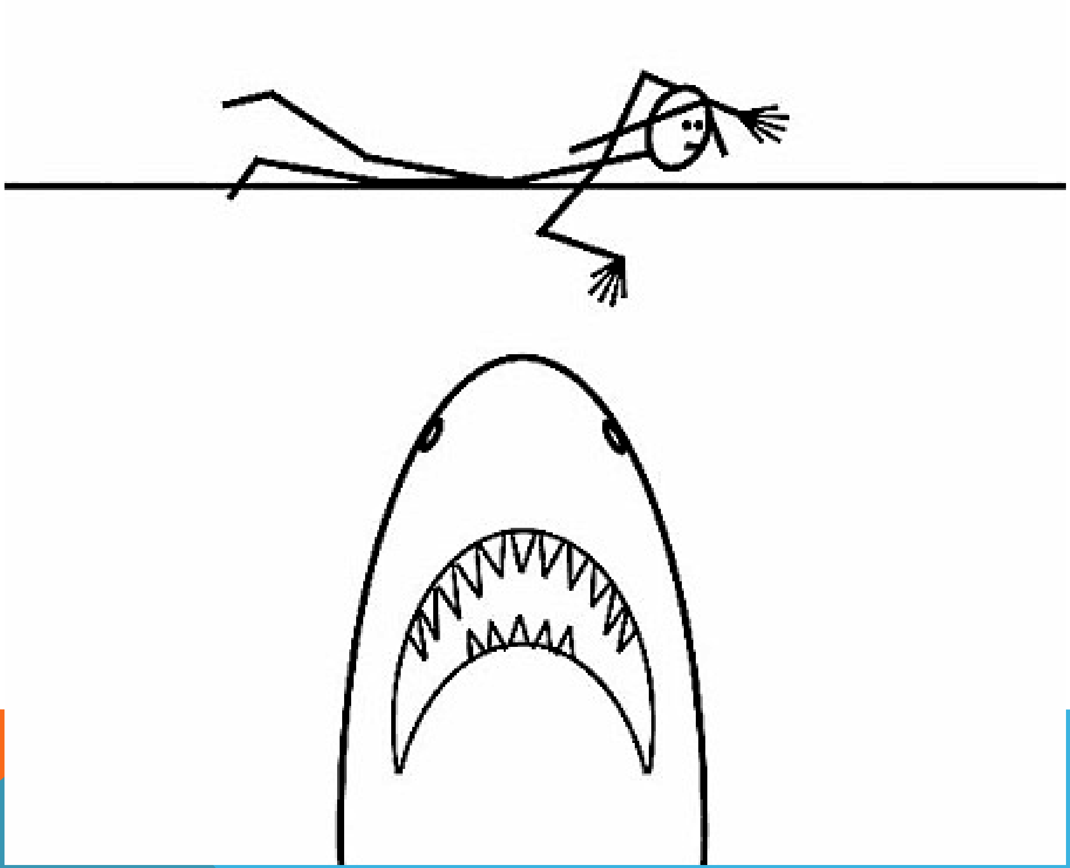
SECURE INTEGRATIONS

- *It is Westfield security direction to protect sensitive data shared between enterprise systems and with external systems through secure (authenticated, access-controlled and auditable) interactions between systems.*

ENTERPRISE SECURITY GOALS / CONSTRAINTS

- Integration to external systems and vendors is through secure, fit-for-purpose, gateways (DataPower, Sterling MFT, WebSEAL) and not directly. Move toward identity federation with vendors and partners.
- Access to Westfield systems and Information from external systems, partners and customers will be controlled through appropriate gateways, never directly to a Westfield system.
- Shifting away from "trusted system" model for internal application integration to a cross-system authentication (XSA) model. Enforce security policy at all levels.
- Security policy enforcement will be externalized (removed) from application code into the XSA framework and middleware. XSA provides the security policy decision points for Westfield's web and services-based systems both internal and externally
- Where needed, enable secure SSO and token-based user (or system) identity propagation to authenticate between systems.
- Wherever possible leverage non-proprietary, extensible standards to enable secure integrations (WS-Security, SAML 2.0 tokens, etc).
- Move away from flat, open network to a segregated network utilizing encryption.

MILESTONE 2…



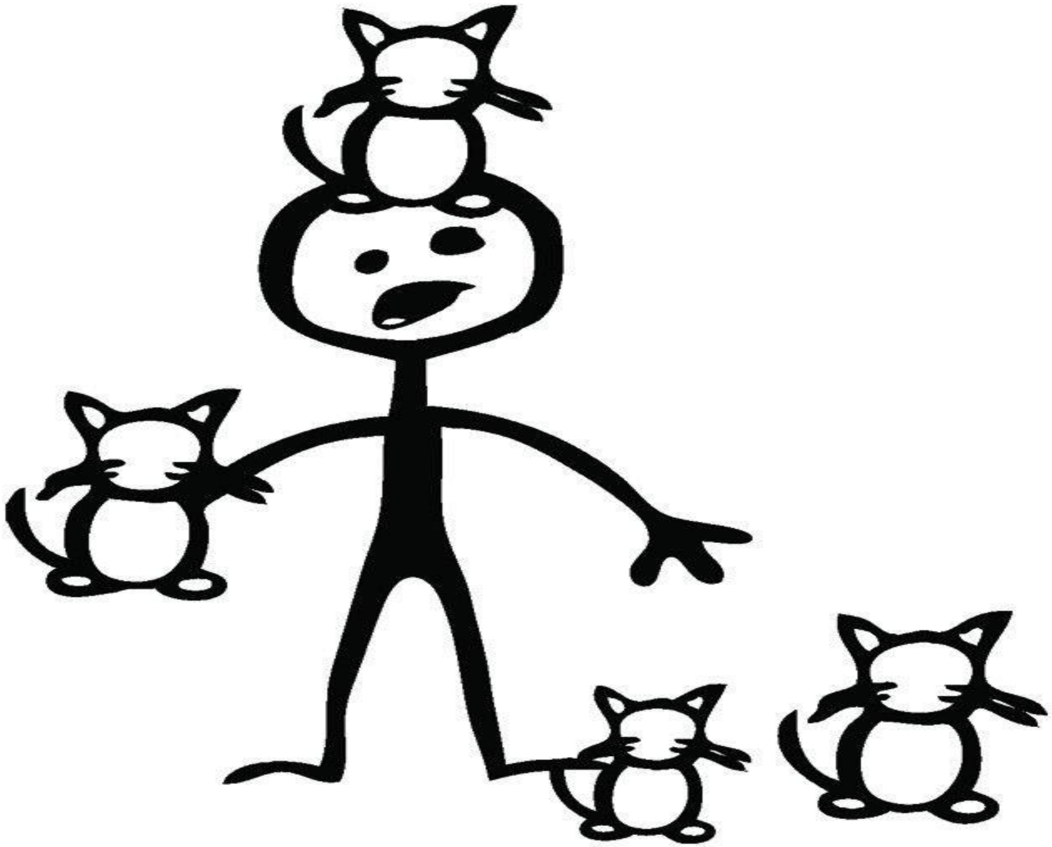
…Security Architecture Roadmap slides go here

;-)

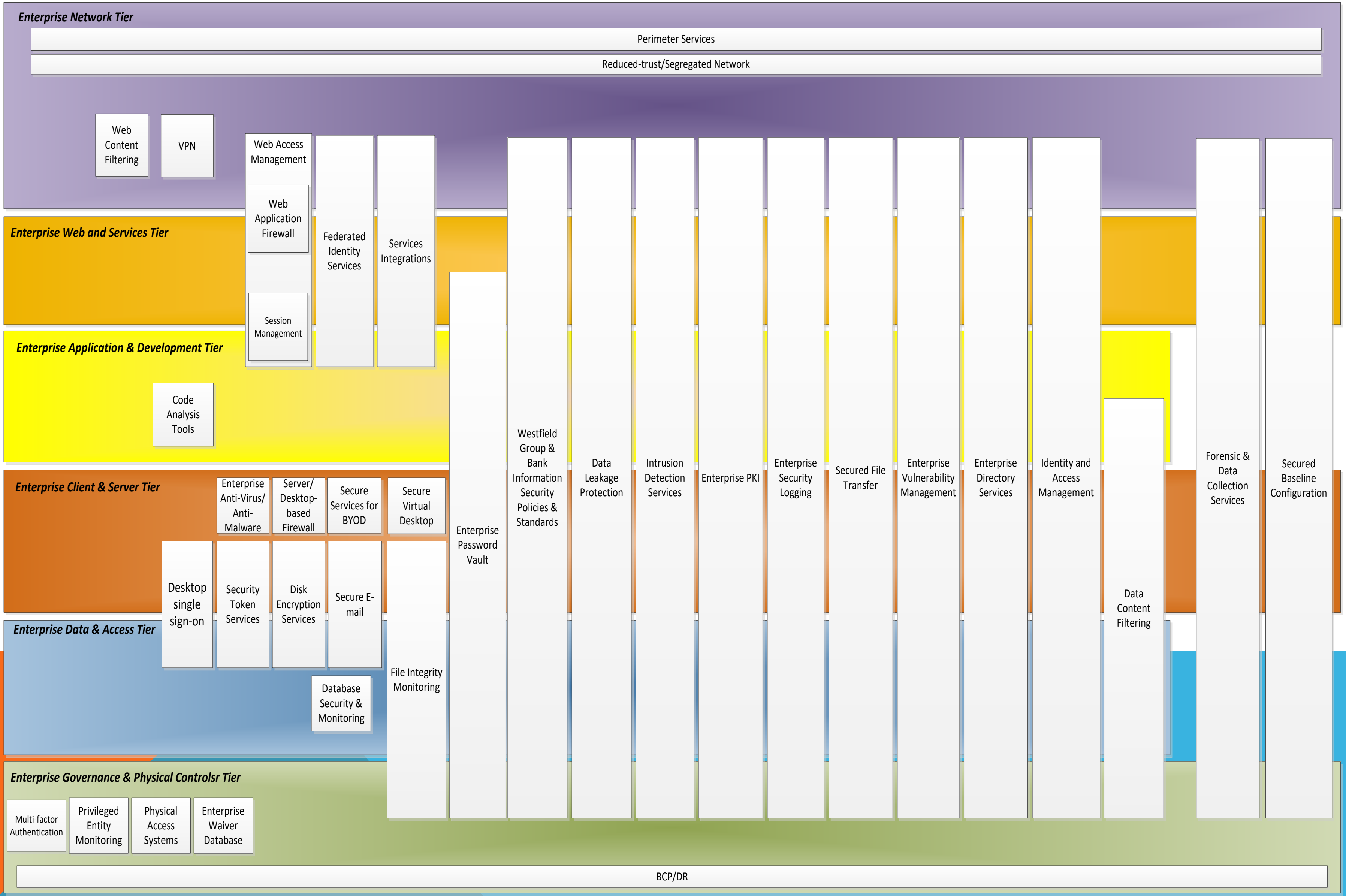
HERDING THE CATS

Unlike a lot of areas in IT or InfoSec, SecArch has a VERY broad reach.

SecArch = Enterprise Architecture, but for security for EVERY platform, system, app



YOU SAY THE WORD “CONTROL” AS IF I HAD ANY…



Enterprise Network Tier

Perimeter Services

Reduced-trust/Segregated Network

Web Content Filtering

VPN

Web Access Management

Web Application Firewall

Session Management

Federated Identity Services

Services Integrations

Enterprise Web and Services Tier

Enterprise Application & Development Tier

Code Analysis Tools

Enterprise Client & Server Tier

Enterprise Anti-Virus/ Anti-Malware

Server/ Desktop-based Firewall

Secure Services for BYOD

Secure Virtual Desktop

Desktop single sign-on

Security Token Services

Disk Encryption Services

Secure E-mail

Enterprise Data & Access Tier

Database Security & Monitoring

File Integrity Monitoring

Enterprise Password Vault

Westfield Group & Bank Information Security Policies & Standards

Data Leakage Protection

Intrusion Detection Services

Enterprise PKI

Enterprise Security Logging

Secured File Transfer

Enterprise Vulnerability Management

Enterprise Directory Services

Identity and Access Management

Data Content Filtering

Forensic & Data Collection Services

Secured Baseline Configuration

Enterprise Governance & Physical Controlsr Tier

Multi-factor Authentication

Privileged Entity Monitoring

Physical Access Systems

Enterprise Waiver Database

BCP/DR

“DISTRIBUTED SECURITY”

A concept is born…..

HOW WE ARE GETTING THERE

APPDEV / SECARCH INITIATIVES

2014 / 2015 Initiatives

- Security and the SDLC
- Scanning Tools
- Security Reference Architecture

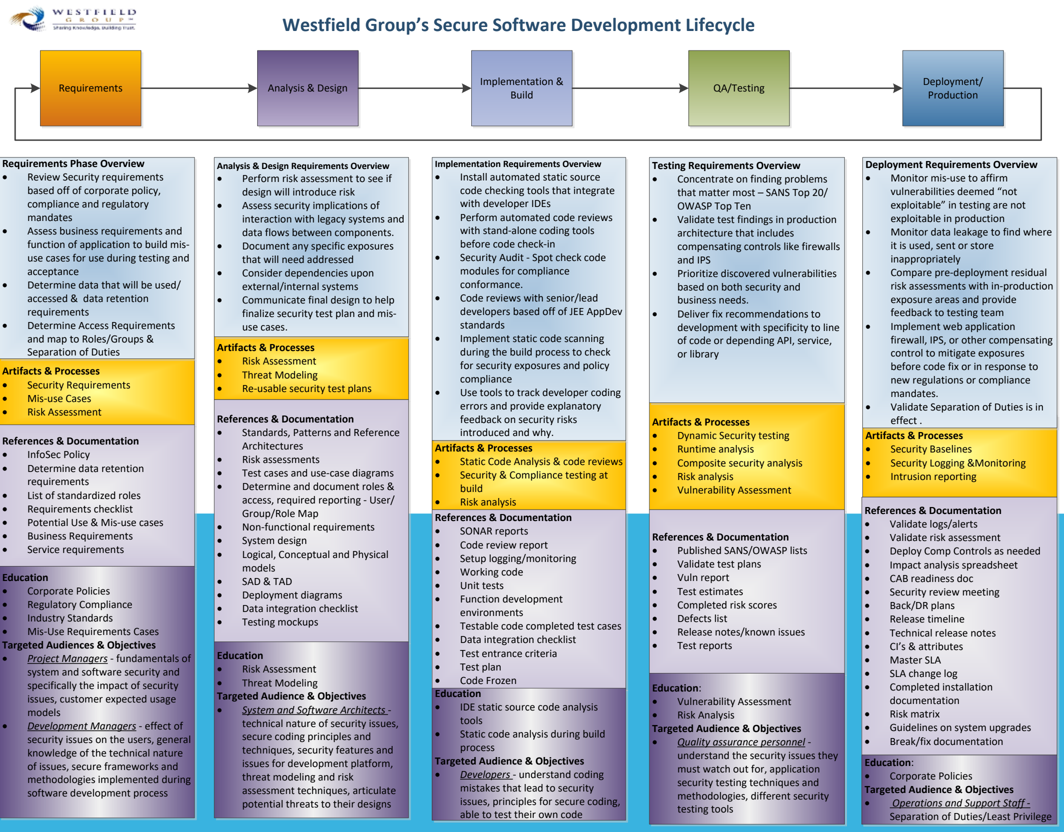
And Beyond

- Certificate Management
- Remediation of existing applications and services

SECURITY AND THE SDLC

✓ SDLC is critical (and not JUST for Development Projects)

- Overview of Tasks
- Artifacts and Processes
- References and Documentation
- Education
- Target Audiences



Westfield Group's Secure Software Development Lifecycle

Requirements → Analysis & Design → Implementation & Build → QA/Testing → Deployment/ Production

Requirements Phase Overview
- Review Security requirements based off of corporate policy, compliance and regulatory mandates
- Assess business requirements and function of application to build mis-use cases for use during testing and acceptance
- Determine data that will be used/ accessed & data retention requirements
- Determine Access Requirements and map to Roles/Groups & Separation of Duties

Artifacts & Processes
- Security Requirements
- Mis-use Cases
- Risk Assessment

References & Documentation
- InfoSec Policy
- Determine data retention requirements
- List of standardized roles
- Requirements checklist
- Potential Use & Mis-use cases
- Business Requirements
- Service requirements

Education
- Corporate Policies
- Regulatory Compliance
- Industry Standards
- Mis-Use Requirements Cases

Targeted Audiences & Objectives
- *Project Managers* - fundamentals of system and software security and specifically the impact of security issues, customer expected usage models
- *Development Managers* - effect of security issues on the users, general knowledge of the technical nature of issues, secure frameworks and methodologies implemented during software development process

Analysis & Design Requirements Overview
- Perform risk assessment to see if design will introduce risk
- Assess security implications of interaction with legacy systems and data flows between components.
- Document any specific exposures that will need addressed
- Consider dependencies upon external/internal systems
- Communicate final design to help finalize security test plan and mis-use cases.

Artifacts & Processes
- Risk Assessment
- Threat Modeling
- Re-usable security test plans

References & Documentation
- Standards, Patterns and Reference Architectures
- Risk assessments
- Test cases and use-case diagrams
- Determine and document roles & access, required reporting - User/ Group/Role Map
- Non-functional requirements
- System design
- Logical, Conceptual and Physical models
- SAD & TAD
- Deployment diagrams
- Data integration checklist
- Testing mockups

Education
- Risk Assessment
- Threat Modeling

Targeted Audience & Objectives
- *System and Software Architects* - technical nature of security issues, secure coding principles and techniques, security features and issues for development platform, threat modeling and risk assessment techniques, articulate potential threats to their designs

Implementation Requirements Overview
- Install automated static source code checking tools that integrate with developer IDEs
- Perform automated code reviews with stand-alone coding tools before code check-in
- Security Audit - Spot check code modules for compliance conformance.
- Code reviews with senior/lead developers based off of JEE AppDev standards
- Implement static code scanning during the build process to check for security exposures and policy compliance
- Use tools to track developer coding errors and provide explanatory feedback on security risks introduced and why.

Artifacts & Processes
- Static Code Analysis & code reviews
- Security & Compliance testing at build
- Risk analysis

References & Documentation
- SONAR reports
- Code review report
- Setup logging/monitoring
- Working code
- Unit tests
- Function development environments
- Testable code completed test cases
- Data integration checklist
- Test entrance criteria
- Test plan
- Code Frozen

Education
- IDE static source code analysis tools
- Static code analysis during build process

Targeted Audience & Objectives
- *Developers* - understand coding mistakes that lead to security issues, principles for secure coding, able to test their own code

Testing Requirements Overview
- Concentrate on finding problems that matter most – SANS Top 20/ OWASP Top Ten
- Validate test findings in production architecture that includes compensating controls like firewalls and IPS
- Prioritize discovered vulnerabilities based on both security and business needs.
- Deliver fix recommendations to development with specificity to line of code or depending API, service, or library

Artifacts & Processes
- Dynamic Security testing
- Runtime analysis
- Composite security analysis
- Risk analysis
- Vulnerability Assessment

References & Documentation
- Published SANS/OWASP lists
- Validate test plans
- Vuln report
- Test estimates
- Completed risk scores
- Defects list
- Release notes/known issues
- Test reports

Education:
- Vulnerability Assessment
- Risk Analysis

Targeted Audience & Objectives
- *Quality assurance personnel* - understand the security issues they must watch out for, application security testing techniques and methodologies, different security testing tools

Deployment Requirements Overview
- Monitor mis-use to affirm vulnerabilities deemed "not exploitable" in testing are not exploitable in production
- Monitor data leakage to find where it is used, sent or store inappropriately
- Compare pre-deployment residual risk assessments with in-production exposure areas and provide feedback to testing team
- Implement web application firewall, IPS, or other compensating control to mitigate exposures before code fix or in response to new regulations or compliance mandates.
- Validate Separation of Duties is in effect .

Artifacts & Processes
- Security Baselines
- Security Logging &Monitoring
- Intrusion reporting

References & Documentation
- Validate logs/alerts
- Validate risk assessment
- Deploy Comp Controls as needed
- Impact analysis spreadsheet
- CAB readiness doc
- Security review meeting
- Back/DR plans
- Release timeline
- Technical release notes
- CI's & attributes
- Master SLA
- SLA change log
- Completed installation documentation
- Risk matrix
- Guidelines on system upgrades
- Break/fix documentation

Education:
- Corporate Policies

Targeted Audience & Objectives
- *Operations and Support Staff -* Separation of Duties/Least Privilege

SCANNING TOOLS

✓ **Helping build better code**

- Static Analysis Scanner (SAS) to find coding problems/security flaws
- Dynamic Analysis Scanner (DAS) to find vulnerabilities in code in runtime environment
- Education

Fun Fact: Cost to fix defect in development: $80. Cost to fix defect in Production: $7,600*

*Source : IBM

SECURITY REFERENCE ARCHITECTURE

✓ Security Patterns Defined

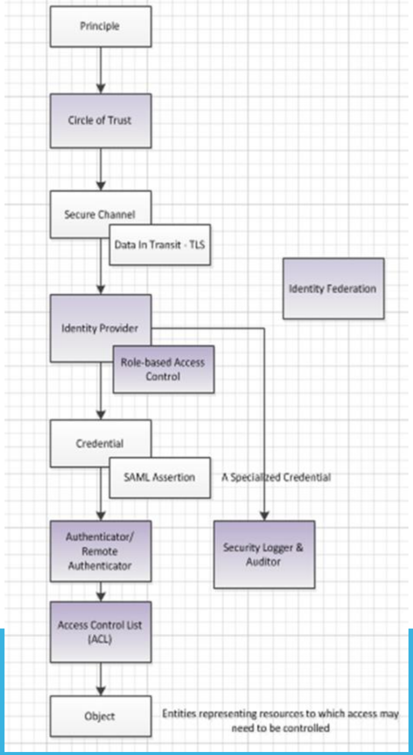
✓ Standards / Guidelines Created

Security Reference Architecture

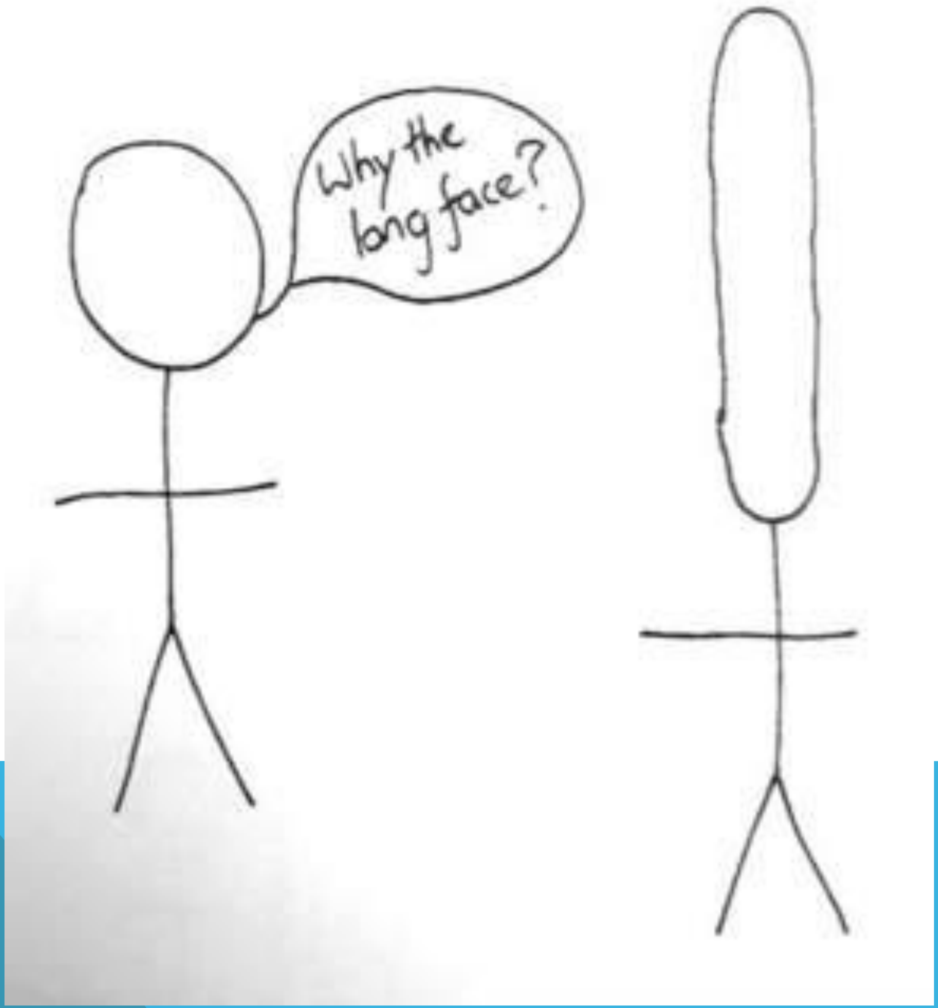
✓ Web Access Management Reference Architecture

Web Service Access Management Reference Architecture

Exceptions documented in Project Solution


Principle
Circle of Trust
Secure Channel
Data In Transit - TLS
Identity Federation
Identity Provider
Role-based Access Control
Credential
SAML Assertion
A Specialized Credential
Authenticator/ Remote Authenticator
Security Logger & Auditor
Access Control List (ACL)
Object
Entities representing resources to which access may need to be controlled

MILESTONE 3



QUESTIONS?