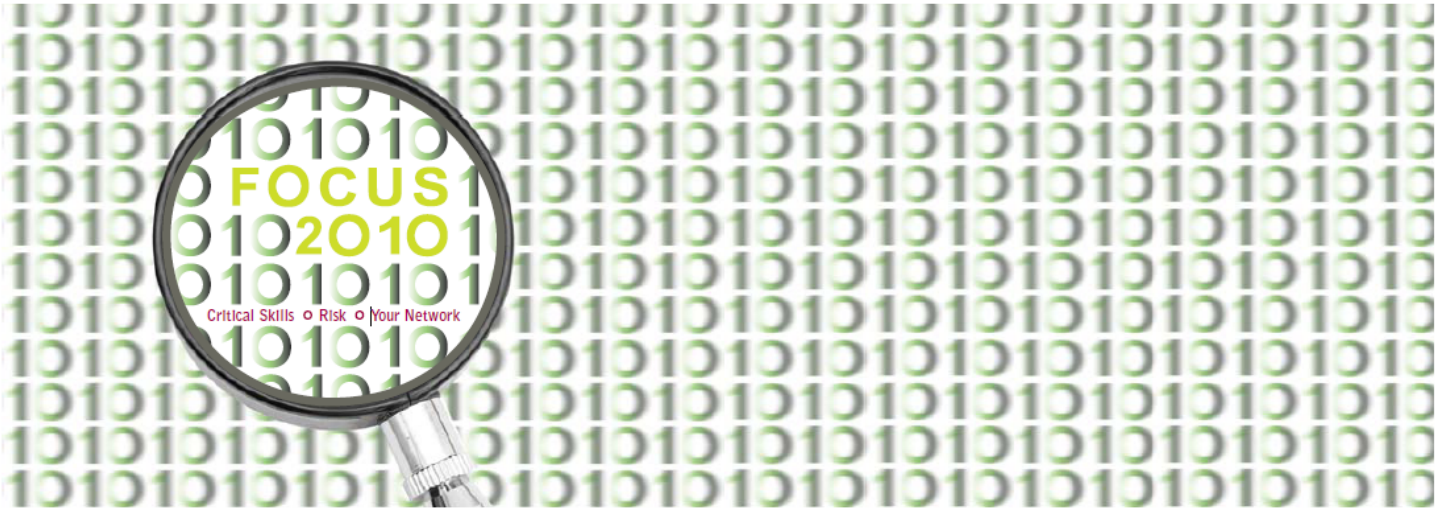
10th Annual SF ISACA Fall Conference

October 4 – 6, 2010



T33: Application Threat Modeling: Evolving Risk Management of Business Applications

Tony UcedaVelez, VerSprite

ISACA®

Trust in, and value from, information systems

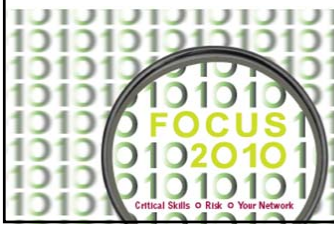
San Francisco Chapter

Threat Modeling Application Environments for Improved Risk Management

ISACA
Trust in, and value from, information systems
San Francisco Chapter

Tony UcedaVélez
Managing Partner
VerSprite

VerSprite
Navigate Beyond Risk



THE PROBLEM

Risk Assessments Losing Credibility

- A lot of theoretical scenarios
  - "If A & B takes place, C could happen"
  - Difficult to provide for accurate quantitative values
  - Control frameworks can't predict the future

Lacks a Hybrid Approach

- Too high level business (pure risk assessment)
- Too techno focused detail (technical risk assessments – vuln scans, static analysis)
- Doesn't integrate to many sources of risk information

Adversarial Approach

- "Us vs. Them" Mentality
- Reduces Criticality of Security Message
- Does not invite collaborative unity toward a single security goal

STATUS QUO FEEDBACK

ASIS/ SIA Risk Assessment Survey (2007)

Question: "Clearly, risk assessment is important, but the real question is this: Are U.S. security practitioners actually conducting risk assessments, and, if so, how are those risk assessments being used? "

ASIS/ SIA Risk Assessment Survey Scope

- Respondents and their organizations represented a broad cross-section of the following industries:
  - agriculture, education, entertainment venues, financial/legal/business professional services, government, health care, hospitality, industrial/manufacturing, information technology/telecommunications/high tech, retail outlets, senior facilities/assisted living, theme parks, warehousing, and many others

ASIS/ SIA Risk Assessment Survey Results

- About one-third of respondents fail to conduct cost-benefit analyses when evaluating options to mitigate risk.
- One-third of security practitioners who perform risk assessments believe their assessments are futile and could not be the basis of a security upgrade.
- Less than half of respondents measure the effectiveness of security systems after installation.
- Between one-third and one-half of respondents do not install security equipment or make other security upgrades in response to a risk assessment.
- About one-third of respondents fail to conduct cost-benefit analyses when evaluating options to mitigate risk.

Bad Press for Security Risk

A SOLUTION

Application Threat Modeling for the Masses

- Builds an attack plan
  - Think like an attacker
  - Conceptualize likely attacks
  - Software Development Life Cycles (SDLC) Integration
  - Migrating from speculative risk scenarios to likely attack vectors

Integrating the What Could Happen ?

- Vulnerability Assessment results reveal areas of weakness
- Pen Testing results provide probabilistic values for exploiting identified vulns
- Static Analysis results for vulnerable code and program objects
- Social Engineering exercises reveals secure unawareness

Integrating the What Does/ Did Happen ?

- Security Incident Data Feeds
- Intrusion Prevention/ Detection Systems
- Firewalls
- Host Based Agents
- Web Application Firewalls (WAFs)

Integrating the What We Got ?

- Security Governance in Action – Finally!
- Policies & Procedures as administrative controls for process related threats
- Standards as countermeasures for application / platform/ network related threats
- Exceptions reveal slightly open 'windows'

Mapping Threat Model Results to Risk Values

- Elevates (legitimizes) probability values
- Incorporate Business Impact Analysis (BIAs) into threat model for quantifying impact
- Provides a tactical scope for application assessments

Threat Modeling Drivers for Building Security In

- Reducing the cost of remediation $$$
- Reducing Knee-Jerk Exception Handling $$
- Introduce Security Awareness as part of OJT $
- Security = > Efficiency $$$

SecuriLocks & the 3 Bears

Taxonomy of Terms

Actors/ Assets (Targets)

- End users that use thick, thin client applications (userID: bsmith, sue.taylor, etc)
- System administrators who regularly interact/ support any part of the application ecosystem
- Achieved via Data Flow Diagramming
- Application accounts used for automated or batched APIs or data interfaces
- Threat modeling terminology lends from Risk Management, Software Development, and IT Architecture

Roles & Privileges

- Rights awarded to pre-defined groups or users for application
- Addresses issues related to impersonation, federated identities in applications
- C.R.U.D analysis (rights to Create, Read, Update, and Delete) across use cases
- Under what security context do you handle report creation, authentication, sensitive transactions, delete account, etc?

Countermeasures

- Equate to controls in risk
- Aimed at mitigating threats and attacks
- Clear injection points for use revealed by threat modeling
- Protection against real risk

Use/ Misuse Cases

- Allows for use cases to be built from functional & security requirements – fat apps are vulnerable!
- Defines branches in attack tree to which attacks, vulns, exploits are correlated
- Defines how the apps can be used & misused
- Business logic flaws finally addressed

Data Flow Diagramming (DFD)

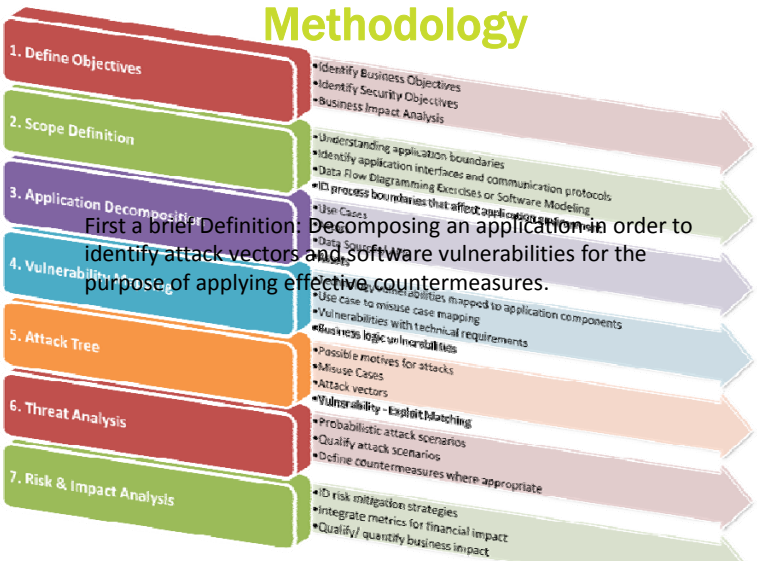
- Steps through the lifecycle of data through an application; application walk through
- Maps out data interfaces across application layers (presentation, app, data, etc)
- Allows for countermeasures to be identified as part of data in transit, while processing, and in storage
- Incorporates actors and assets as data flow start & end points

Trust Boundaries

- Boundaries that define where trust should be granted and to what degree
- Allows for the consideration of new threats (privilege escalation, etc) and countermeasures (authentication controls) that relate to trust amongst application calls

Methodology

Methodology



First a brief Definition: Decomposing an application in order to identify attack vectors and software vulnerabilities for the purpose of applying effective countermeasures.

Thank You!

Threat Modeling Methodology Myths

- No widely accepted methodology exists today.
- By widely, we simply mean no organization has defined and patented a threat modeling
- STRIDE & DREAD are not methodologies, threat and risk classifications respectively

Key Components to Threat Modeling

- Steps 3,4,5,6 equate to 'secret sauce'
- Step 3: App Decomposition allows for greater understanding of app to all involved parties (threat modeler, developers, architects, sys admins)
- Step 4: Vuln Mapping integrates unmanaged vulnerabilities in order to ID a window for an exploit. Something to worry about.
- Step 5: Attack Tree evolves beyond the theoretical to lets let our guys try to exploit this
- Step 6: Threat Analysis shows the net effect of vulns * attacks - countermeasures

What Threat Modeling is NOT

Beyond The Hype

- As with any new buzz in security, its not long before a good thing mutates in meaning and application
- Not a replacement for risk assessments
  - Risk assessments have their place for ongoing risk analysis of deployed application environment
  - Still the preferred choice for vendor applications (tough to build a detailed threat model on vendor application environments)
  - Risk assessments benefit from threat modeling deliverables for an improved and targeted risk analysis

Threat Modeling Distortion

- Not a loosely defined exercise to complete a check box
- An attack tree does is not a threat model
- A data flow diagram (DFD) is not a threat model
- Breaking up bits and pieces of the threat modeling methodology is just that – a broken or incomplete threat model

Not Another Silver Bullet

- Aimed at elevating the predictive nature of risk analysis by understanding viable threats and attack patterns for apps
- Still warrants and depends on auxiliary processes and disciplines across security, compliance, and IT
  - Vuln mgt,
  - Business impact analysis,
  - Security governance (policy/ standard mgt),
  - Incident analysis & response,
  - DLP solutions,
  - Network Operations
- Requires a collaborative work environment
  - Barriers to information gathering poses a problem

Facets of Threat Modeling

Threat Modeling ACME Company

- SDLC Efforts – Define & Design Time
  - PMs, business analysts, business owners devise functional requirements (Definition Phase)
  - Architects and IT Leaders speak to architectural design and platform solutions (Design Phase)
  - Governance leaders inject compliance & standards requirements for during he design phase; BIA
  - Threat Model* (SOC/ NOC fed), DFDs Introduced, Trust Boundaries defined, Countermeasures proposed

ACME Example – Dev Time

- Time to Write the Code - Development
  - Incorporates both functional & security reqs.
  - Developers now more aware of potential threats
- Countermeasures developed within applications
  - Validation Checks
  - Reduced Privs
  - Proper encoding techniques

Be the BlackHat

- Discovering Vulns & Applying Attacks
- QA tests functional features; scope creep in use cases
- Threat modeler tests for vulns, exploit opportunities, config flaws, logic flaws, bad design
- QA can serve as security testing group
  - Rising trend to leverage QA
- Sanely be Dr. Jekyl/ Mr. Hyde

Threat Identification & Impact

- Enumerate the threats to the application elements
  - PII theft
  - IP theft
  - Sabotage driven threats
  - Malware upload
- Identify the impact for the most likely attack vectors
  - Social engineered emails
  - Web Forms/ Fields
  - Email related auxiliaries uses to web apps
  - Other data interfaces supporting web application environment

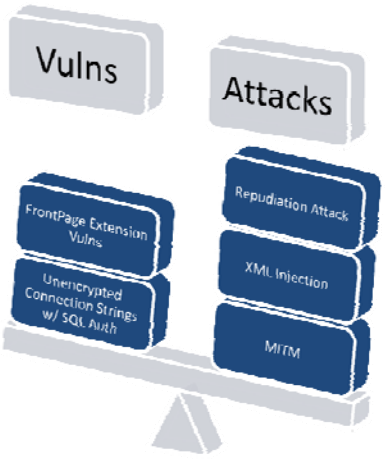
Use cases/ vulns beget misuse cases

- Every function has a potential dysfunction; need to enumerate and test application functions
- Listing of vulns for mapping can originate from subscribed vulnerability feeds/ vulnerability signatures from vendors
- Some Sources: SecurityFocus, US-CERT, Symantec, Microsoft
- Map vulns to employed platforms and software technologies
- Attack tree begins to take shape

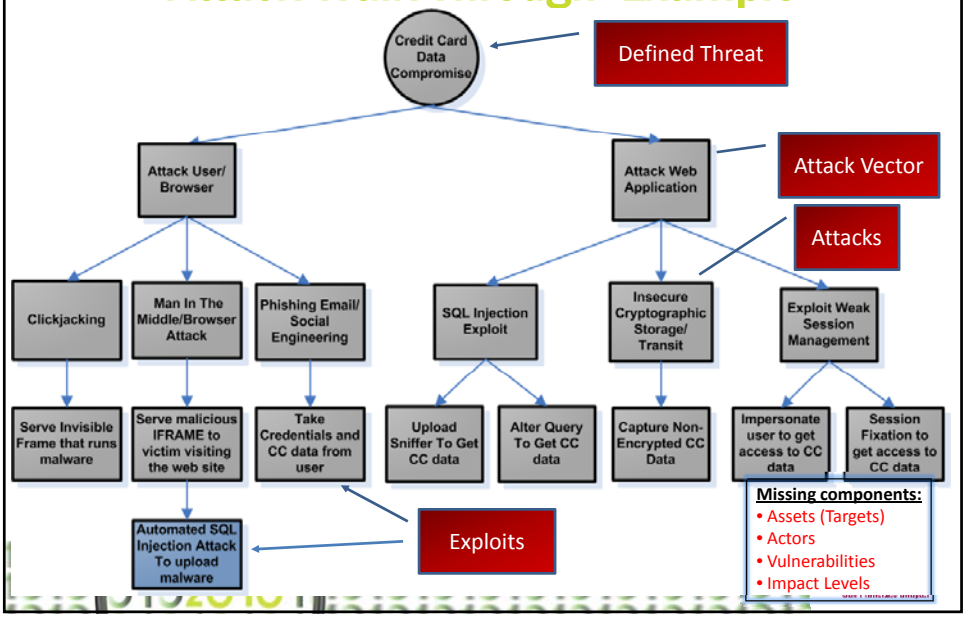
Mapping Use Cases to Misuse Cases

Misuse cases/ vulns beget exploits

- Exploitation is the proof. We all need proof.
- Given time constraints, partial exploits may be acceptable; educating that attacks are layered.
- Exploitation may address identified vulns, business logic flaws, and/ or non-published vulnerabilities



Attack Walk Through Example



Data Flow Diagramming Exercises

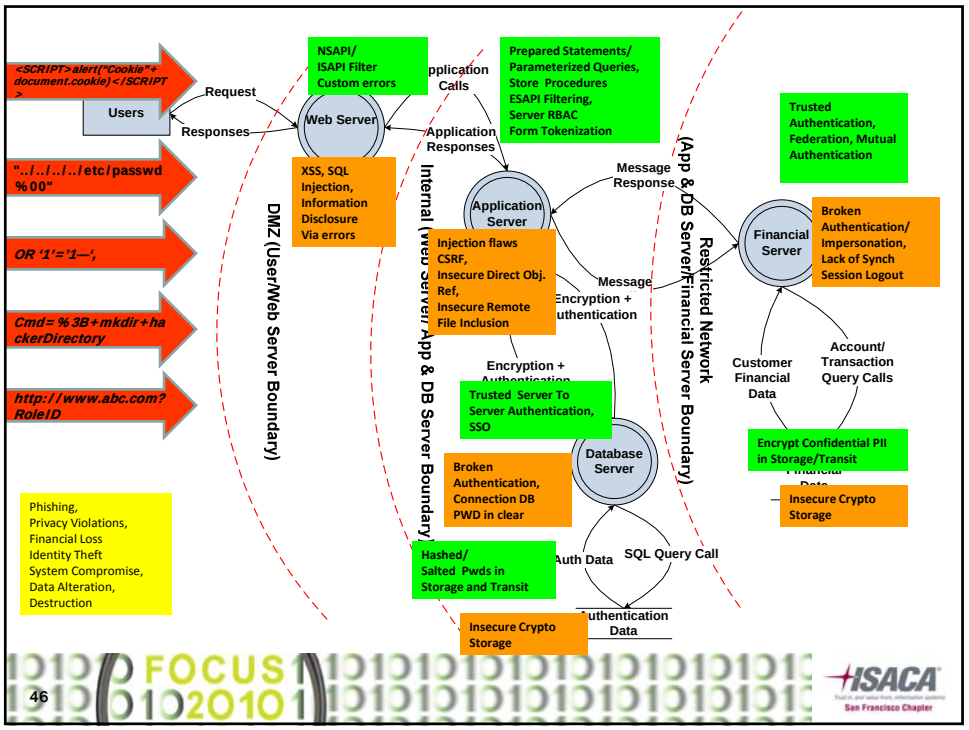
- Identify entry and exit points as well as related access levels
  - Internal and external interfaces
  - What are the trust boundaries?
  - Single/ Cross Domain traversals
  - Mapping out Networks

Exploits beget countermeasures

- Unacceptable risks give way to countermeasure development
- Develop countermeasures based upon the net risk of an application environment at multiple levels
  - Baseline configuration
  - Design and programmatic controls
  - 3rd party software/ COTS

Countermeasures

- Identify mitigations to the previously identified attacks-to-vuln relationship by locating the countermeasures
  - Native configuration countermeasures
  - ESAPI encryption (web.config)
  - TCP Wrappers
  - Mod Security
  - HTTPS/ HTTP validation
- Develop new countermeasures



Drivers & Value-Add

- Remediation takes place for risky findings
  - Understanding threats catalyzes remediation
- Abides by Building Security In concept
- Improves software assurance model
- Cost/ Time savings stem from time savings across multiple efforts
  - Chg Mgt, Post Implementation Security Testing, Exception Management

What does Risk mean anymore?

Do We Know Real Risk?

- Leaders have become desensitized to risk; its meaning has warped into opinionated thought exercises
  - Risk = ((Threats (probability) * Vulnerability)/Countermeasures) * Impact
  - Impact assumes threat will take place
  - Impact = # of occurrences * SLE
  - Occurrences may equate to incidents (records lost, number of servers, etc)
  - SLE = Exposure factor * Asset value

Why Management Doesn't Care

- Data rarely is relatable to business or operational impact
- Either too technical or too high level.
- Instead presents a laundry list of remediation items – more work!

Metrics, Research, & Threat Modeling

- Building Security In: A new risk modeling paradigm for developing applications
- Case & Point: Demonstrating how attack happen (pen test results, dynamic analysis, static analysis)
- Understanding Threats: Incorporates threat feeds, network traffic logs, intrusion attempts

Financial Incentives to Threat Modeling