28TH ANNUAL PACIFIC NW SOFTWARE QUALITY CONFERENCE
OCTOBER 18-20, 2010

McAfee

Adaptive Application Security Testing Model

Ashish Khandelwal
Gunankar Tyagi

Agendum

- *What is Security Testing & its relevance ?*
- *Traditional Security Testing Approach*
- *The Adaptive Application Security Testing Model*
- *Case Study – Application Product Testing Success Story*
- *Q&A*

Confidential McAfee Internal Use Only

Security Testing

- *A Quick Glance*
- *Security V/s Functional Testing*
- *Myths*

Confidential McAfee Internal Use Only

Cost Impact

€

Monster.com suffered a heavy security breach in Aug 2007 that reportedly resulted in the theft of the confidential information for some 1.3 million job seekers.

$

元

The TJX Company breach, which was first reported in January of 2007, has been widely recognized as the largest reported theft of personal details ever lost by a company.

Operation Aurora affected as many as 2,411 companies and compromised data ranges from intellectual property, classified documents to credit card transaction details

₹

£

¥

LOVE BUG exploited Microsoft Outlook e-mail client to execute programs. The damage resulting from this virus was reported to be in the billions of dollars.

Confidential McAfee Internal Use Only

Rich Man's Wisdom

A man who wants to remain rich , will make sure he locks his money in

The "VAULT"

Confidential McAfee Internal Use Only

Case Study – Product Context Setting

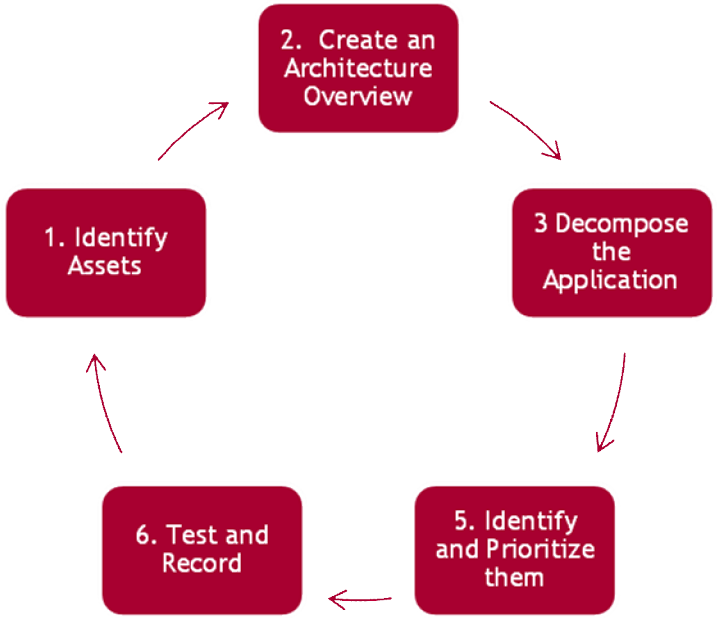
- *Application Product team*
- *Integrated Engineering team*
- *Skill Sets*
- *Specialized Skilled*

Confidential McAfee Internal Use Only

Threat Model and its constraints

GET SET GO

PROCESS FLOW



CONSTRAINTS

Complexity

- *Approach is time consuming as it involves listing and analyzing every possible asset in the product*

Connectivity

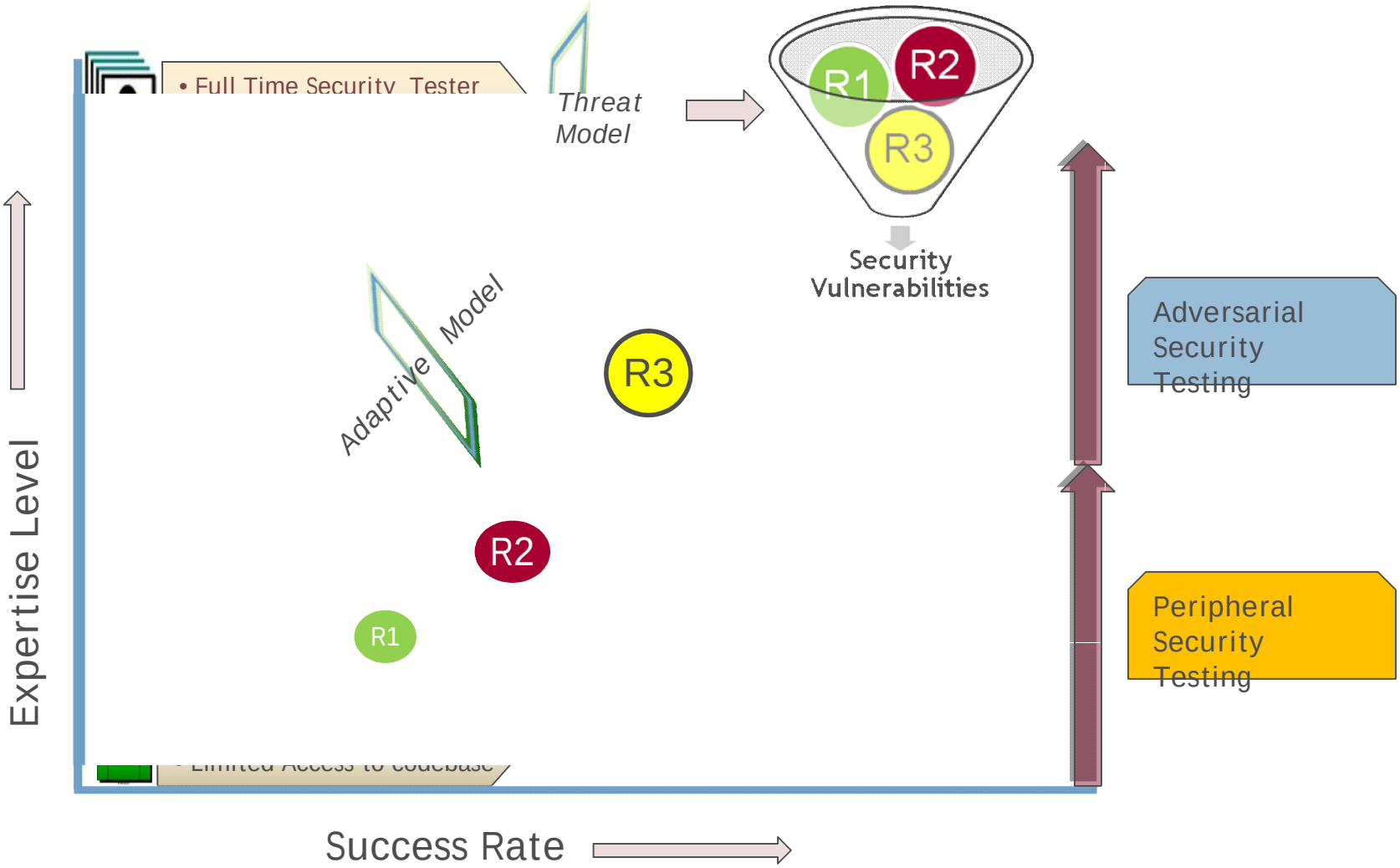
- *Access to architectural diagram & specifically code is not always possible.*

Changeability

- *Agile methodology & environmental exposure can endanger threat model base.*

Afee Internal Use Only

Adaptive Ladder



Confidential McAfee Internal Use Only

Adaptive Model - Highlights

Two-tier sequential model

Peripheral Security Testing (PST)
Adversarial Security Testing (AST)

Each of these testing types is defined in terms of

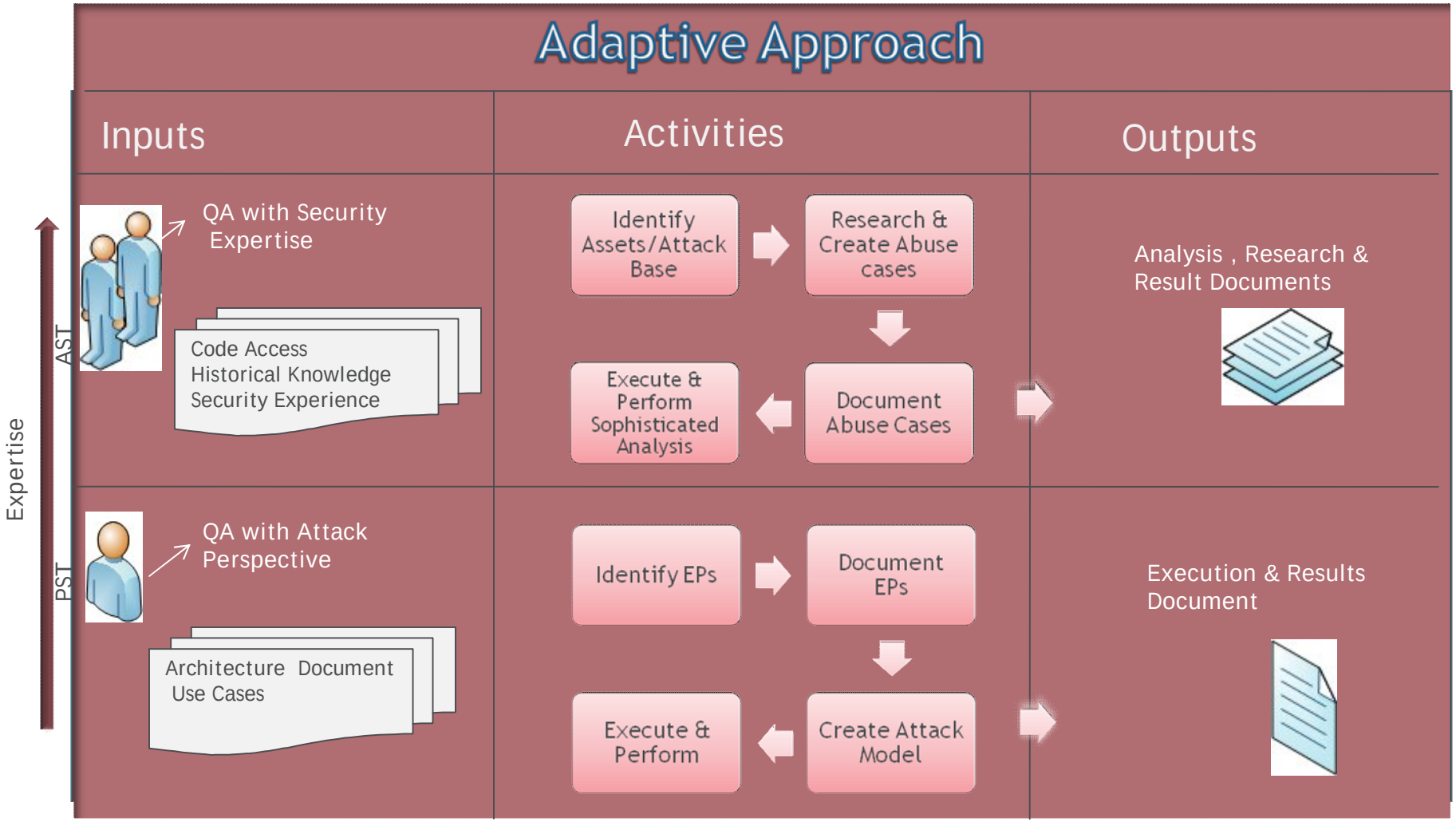
✓Inputs ✓Activities ✓Outputs

Adaptive Model kicks off with PST and then helps to

✓Enhance security knowledge and experience
✓Constantly Deliver results
✓Build perquisite for AST

Confidential McAfee Internal Use Only

Adaptive Model – Basic Workflow



AST – Adversarial Security Testing
PST – Peripheral Security Testing
EPs – Entry/Exit Points

Confidential McAfee Internal Use Only

Peripheral Security Testing

McAfee

Entry Points

Place where inputs are supplied to your application

Exit Points

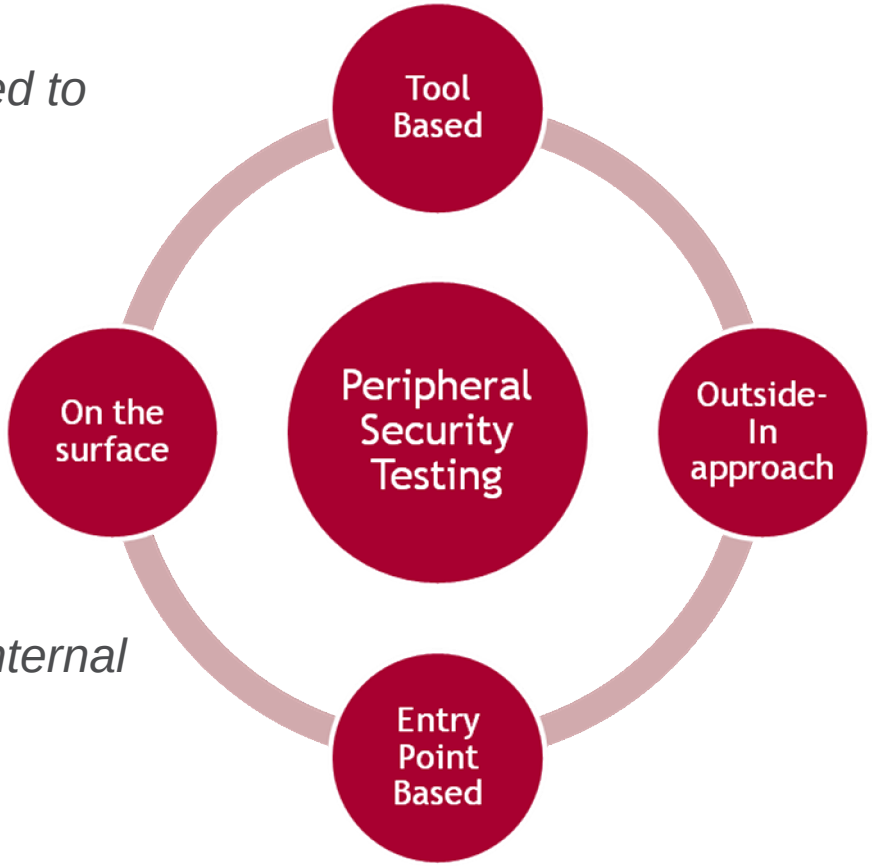
Desirable/ undesirable output from the application.

Outside Approach

Without much knowledge of internal implementation

On the Surface

Easier to detect and require less effort.



Confidential McAfee Internal Use Only

Sample Study I (Peripheral Security Testing)

McAfee

PROCESS FLOW

List Entry Points
- Files, Folders
- Registry Entries

Prepare Attack Model
- Information Disclosure
- Weak Permissions
- Buffer Overflow

Use Tools & Scripts
- FileMon
- Regmon
- ACL Editor

Execute & Record
- Write Results
- Document Issues

ATTACK MODEL

Does your product functionality hamper if you deny the permissions to the temp folder ?

Do the files(logs/event xml/binaries) contain sensitive data ?

Is there a way in which you can cause buffer overflow in the file extension /file names?

Confidential McAfee Internal Use Only

Sample Study II (Peripheral Security Testing)

McAfee

PROCESS FLOW

List Entry Points

- *Named Pipes (IPC)*

Prepare Attack Model

- Exploit weak permission
- Hijack the creation
- Impersonate the client

Use Tools & Scripts

- PipSec
- PipeList
- CreateAgentPipe
- ObjSD

Execute & Record

- Write Results
- Document Issues

ATTACK MODEL

Identify Named Pipes (pipe list)

View Permissions of Named Pipe (ObjSD)

Check the product against Hijacking or Impersonating the Named pipe

Confidential McAfee Internal Use Only

Peripheral Security Testing Checklist

S.No.	Entry/Exit Points	Attack Model	Tools/Scripts
1	File & Folders	Information Disclosure Weak Permissions Buffer Overflow	FileMon ACL Editor strings
2	Sockets	Man-in-the-middle Attack Sniffing network traffic Send malicious data	Wire shark netstat.exe netcat
3	Registry Entries	Registry Accessed by the product Permission of the registry keys	Regmon ACL Editor
4	Named Pipes	Exploit weak permission Hijack the creation Impersonate the client	PipSec PipeList CreateAgentPipe ObjSD
5	User Interfaces	Shatter Attack Format String Attacks	Shatter Tool WebText Convertor
6	Command Line Arguments	Exploit Undocumented command Line switches	Command Line switches /?, -?, /h, or -h. Process Explorer Image tab
7	Environment Variables	Uncover Environment Variables used by Product Manipulating data inside Product defined Environment Variables	Process Explorer Environnent Tab System Environnent Variable Tab
8	ActiveX Control	ActiveX Repurposing Attacks ActiveX Fuzzing	COMRaider OLEView
9	Drivers	I/O Verification Deadlock Detection Dangerous APIs Exceptions/Handlers/Memory Loading and Unloading Filter Driver Attach and Detach Filter Driver	Windows Utility-> Verifier.exe Windows Utility -> fltmc Microsoft Application Verifier Velocity Tool by Microsoft

Confidential McAfee Internal Use Only

Adversarial Security Testing

McAfee

Attack Base

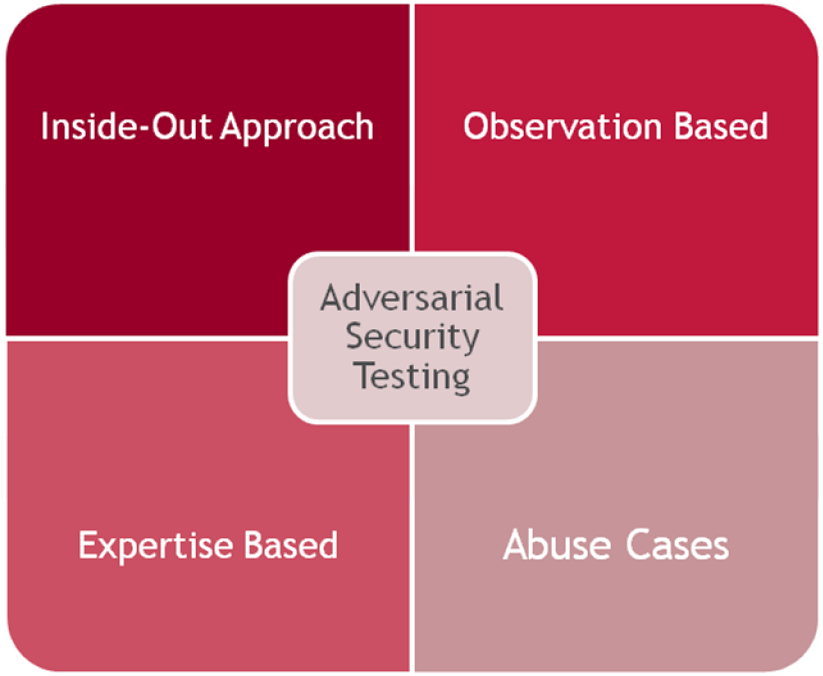
An entity of the product or the Operating System which can be Manipulated to perform an attack on software.

Observation Based

Gather past vulnerability information About the attack base.

Abuse Cases

Abuse cases (sometimes called misuse cases as well) are a tool that can help you begin to think about your software the same way that attackers do.



Confidential McAfee Internal Use Only

Sample Study III(Adversarial Security Testing)

McAfee

ATTACK MODEL

- ✓ Complexity of ACL's configuration
- ✓ Permissions cannot be assigned to all objects
- ✓ Exploiting Integrity Level (Vista specific)

PROCESS FLOW

Identify Attack Base
- Access Control List

Research
- ACL and Code Base
- Historical findings

Abuse Cases
- Coding Error
- Privilege Escalation
- Denying Permissions

Execute & Record
- Result Analysis
- Document Issues

Confidential McAfee Internal Use Only

Adversarial Security Testing Checklist

S.No.	Attack Base	Abuse Case Scenarios	References and Historical Knowledge
1	Access Control List	Verification of apt ACL's for your product resources Target NULL DACL Look for dangerous ACE types -> Everyone (WRITE_DAC) -> Everyone (WRITE_OWNER) -> Everyone (FILE_ADD_FILE) Target Windows DAC weakness Target Windows MIC weakness	Shatter Attack http://www2.packetstormsecurity.org/cgi-bin/search/search.cgi?searchtype=archives&counts=26&searchvalue=win2000+attack+.c Exploiting Integrity Levels http://archive.hack.lu/2007/cracking_windows_access_control.ppt
2	Shell Extensions	List out shell extensions used by your product List the resources utilized by our Shell Extension Behavior of shell extension. Effect of impersonating your product shell extensions.	http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-5902
3	Plugins	List resources used by BHO Learn how to write a BHO Understand functionality of IE Plugin. -> This can give more attack vectors Effect of impersonating our product BHO Find a way to disable IE Plugin	http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2004-2382
4	Denial Of Service	Do basic analysis of DOS Attacks Identify the services rendered by your product Identify the ports used by the services Identify tools to send specially crafter packets to perform a DOS Attack on our product. (use historical info) Analyze the results	http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-1855

Confidential McAfee Internal Use Only

AASTM Model - Recap

- Position yourself on the Adaptive Ladder and then design your security testing strategy.
- The idea is to find security defects in the product. A model is important but not a constraint
- Follow Peripheral and Adversarial approaches as a guideline to target security flaws.
- Creating a dedicated Skill-set base within the team helps a lot.
- Even if it's an ad-hoc approach it's good to expose some security shortcomings

Confidential McAfee Internal Use Only

References

- Tom Gallagher, Bryan Jeffries and Lawrence Landauer , Microsoft Press Hunting Security Bugs, 2006
- G. Hoglund and G. McGraw, Exploiting Software, Addison-Wesley, 2004.
- http://news.cnet.com/2100-1001-240112.html
- http://en.wikipedia.org/wiki/Operation_Aurora
- http://www.infosecwriters.com/text_resources/pdf/need_for_security_testing.pdf
- http://www.zdnet.co.uk/news/it-strategy/2007/11/14/the-worst-it-security-incidents-of-2007-39290745/
- http://en.wikipedia.org/wiki/Shatter_attack
- http://en.wikipedia.org/wiki/Mandatory_Integrity_Control
- http://archive.hack.lu/2007/cracking_windows_access_control.ppt
- http://nikkigsblog.files.wordpress.com/2010/04/locked-house.jpg
- http://msdn.microsoft.com/en-us/library/ff648644.aspx

Confidential McAfee Internal Use Only

Q & A

For any help/query

Ashish_Khandelwal@Mcafee.com

Gunankar_Tyagi@McAfee.com

Confidential McAfee Internal Use Only