The PMC Group LLC

Engineering a better tomorrow today

NIST SP 800-82 Rev 2
ICS Overlay
and
Key Security Controls

January 28, 2014

National Institute of
BUILDING SCIENCES

An Authoritative Source of Innovative Solutions for the Built Environment

NIST SP 800-82 Rev 2 ICS Key Security Controls

Inventory
- CM-8 Information System Component Inventory
- PM-5 Information System Inventory
- PL-7 Security Concept of Operations
- PL-8 Information Security Architecture
- SC-41 Port and I/O Device Access
- PM-5 Information System Inventory

Central Monitoring
- AU-6 Audit Review, Analysis, and Reporting
- CA -7 Continuous Monitoring
- IR-5 Incident Monitoring
- IR-6 Incident Reporting
- PE-6 Monitoring Physical Access
- PM-14 Testing, Training and Monitoring
- RA-5 Vulnerability Scanning
- SC-7 Boundary Protection
- SI-4 Information System Monitoring
- SI-5 Security Alerts, Advisories, and Directives

Test and Development Environment
- CA-8 Penetration Testing
- CM-4 Security Impact Analysis
- CP-3 Contingency Training
- CP-4 Contingency Plan Testing and Exercises
- PM-14 Testing, Training and Monitoring

Critical Infrastructure
- CP-2 Contingency Plan
- CP-6 Alternate Storage Site
- CP-7 Alternate Processing Site
- CP-10 Information System Recovery and Reconstitution
- PE-3 Physical Access Control
- PE-10 Emergency Shutoff
- PE-11 Emergency Power
- PE-12 Emergency Lighting
- PE-13 Fire Protection
- PE-14 Temperature and Humidity Controls
- PE-17 Alternate Work Site
- PM-8 Critical Infrastructure Plan

Acquisition and Contracts
- AU-6 Audit Review, Analysis, and Reporting
- CA -7 Continuous Monitoring
- SA-4 Acquisitions
- PM-3 Information System Resources
- PM-14 Testing, Training and Monitoring

ICS Overlay CM-4

CM-4 SECURITY IMPACT ANALYSIS

Control: The organization analyzes changes to the information system to determine potential security impacts prior to change implementation.

Control Enhancements:

(1) *SECURITY IMPACT ANALYSIS | SEPARATE TEST ENVIRONMENTS*
The organization analyzes changes to the information system in a separate test environment before implementation in an operational environment, looking for security impacts due to flaws, weaknesses, incompatibility, or intentional malice.

ICS Overlay CA-8

CA-8 PENETRATION TESTING

Control: The organization conducts penetration testing [*Assignment: organization-defined frequency*] on [*Assignment: organization-defined information systems or system components*].

ICS Supplemental Guidance: Penetration testing is used with care on ICS networks to ensure that ICS functions are not adversely impacted by the testing process. In general, ICS are highly sensitive to timing constraints and have limited resources. Example compensating controls include employing a replicated, virtualized, or simulated system to conduct penetration testing. Production ICS may need to be taken off-line before testing can be conducted. If ICS are taken off-line for testing, tests are scheduled to occur during planned ICS outages whenever possible. If penetration testing is performed on non-ICS networks, extra care is taken to ensure that tests do not propagate into the ICS network.

ICS Overlay CP-3

CP-3 CONTINGENCY TRAINING

Control: The organization provides contingency training to information system users consistent with assigned roles and responsibilities:
a. Within [*Assignment: organization-defined time period*] of assuming a contingency role or responsibility;
b. When required by information system changes; and
c. [*Assignment: organization-defined frequency*] thereafter.

Control Enhancements:

(1) *CONTINGENCY TRAINING | SIMULATED EVENTS*
The organization incorporates simulated events into contingency training to facilitate effective response by personnel in crisis situations.

ICS Overlay CP-4

CP-4 CONTINGENCY PLAN TESTING AND EXERCISES

(2) *CONTINGENCY PLAN TESTING | ALTERNATE PROCESSING SITE*
The organization tests the contingency plan at the alternate processing site:
(a) To familiarize contingency personnel with the facility and available resources; and
(b) To evaluate the capabilities of the alternate processing site to support contingency operations.

ICS Overlay PM-14

PM-14 TESTING, TRAINING, AND MONITORING

Control: The organization:
a. Implements a process for ensuring that organizational plans for conducting security testing, training, and monitoring activities associated with organizational information systems:
1. Are developed and maintained; and
2. Continue to be executed in a timely manner;
b. Reviews testing, training, and monitoring plans for consistency with the organizational risk management strategy and organization-wide priorities for risk response actions.

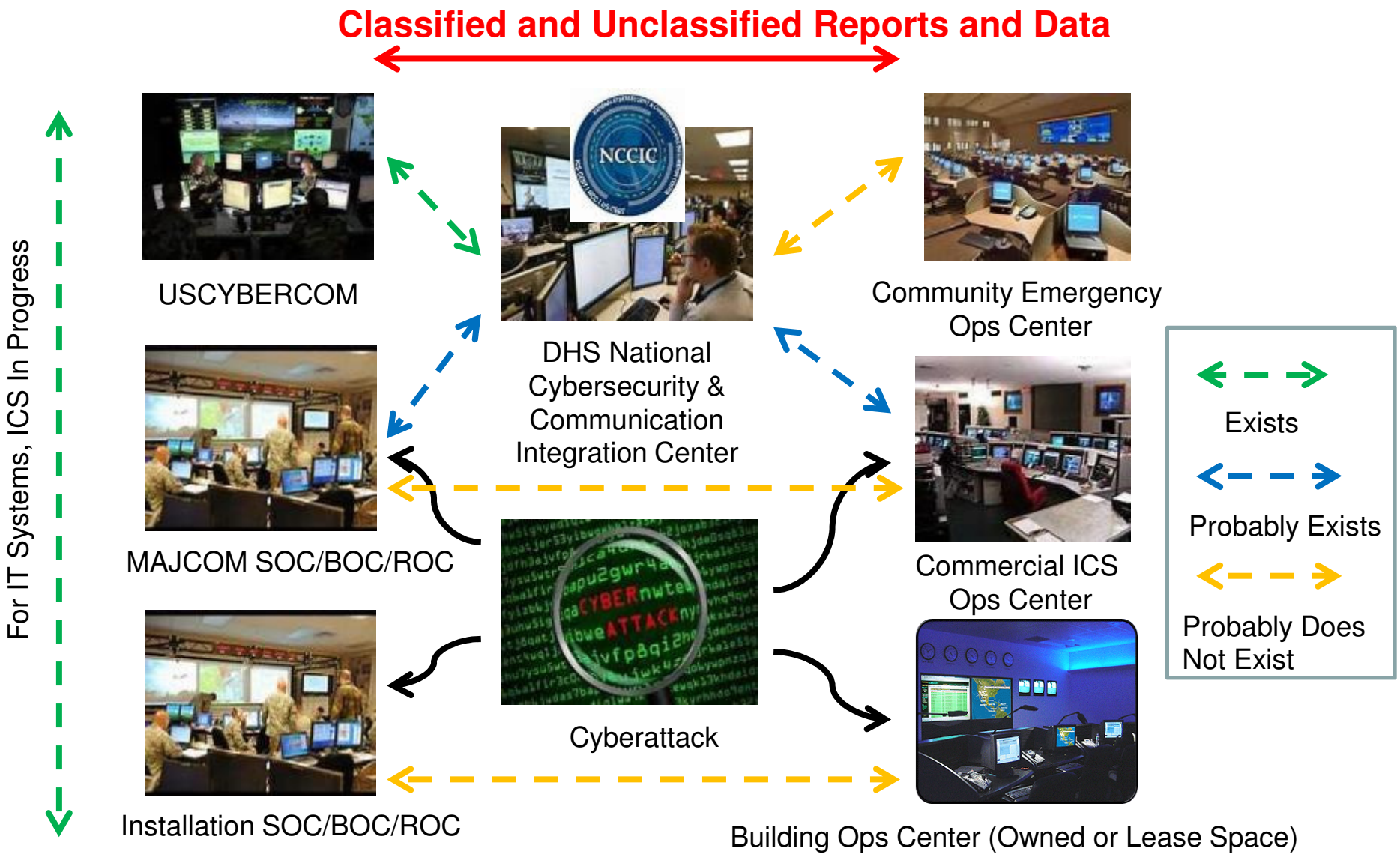
OWASP Top 10 Risks 2013

Top 10 Risk Factor Summary

The following table presents a summary of the 2013 Top 10 Application Security Risks, and the risk factors we have assigned to each risk. These factors were determined based on the available statistics and the experience of the OWASP Top 10 team. To understand these risks for a particular application or organization, you must consider your own specific threat agents and business impacts. Even egregious software weaknesses may not present a serious risk if there are no threat agents in a position to perform the necessary attack or the business impact is negligible for the assets involved.

RISK	Threat Agents	Attack Vectors	Security Weakness		Technical Impacts	Business Impacts
		Exploitability	Prevalence	Detectability	Impact	
A1-Injection	App Specific	EASY	COMMON	AVERAGE	SEVERE	App Specific
A2-Authentication	App Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App Specific
A3-XSS	App Specific	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	App Specific
A4-Insecure DOR	App Specific	EASY	COMMON	EASY	MODERATE	App Specific
A5-Misconfig	App Specific	EASY	COMMON	EASY	MODERATE	App Specific
A6-Sens. Data	App Specific	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	App Specific
A7-Function Acc.	App Specific	EASY	COMMON	AVERAGE	MODERATE	App Specific
A8-CSRF	App Specific	AVERAGE	COMMON	EASY	MODERATE	App Specific
A9-Components	App Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App Specific
A10-Redirects	App Specific	AVERAGE	UNCOMMON	EASY	MODERATE	App Specific

Conceptual CI and ICS Information Sharing



ICS Test and Development Environment

A Test and Development Environment is essential to:

- Perform Patch and Vulnerability Management
- Run the scanning and intrusion detection systems on test ICS cases before deployment onto production systems
- Conduct penetration testing of new devices, software, and configurations before deployment onto production systems
- Develop flow down procedures of Whitelists
- Create the Firewall Rule Sets and test configuration
- Perform Audit and Recovery analysis
- Practice area for the COOP CONOPS

ICS Test and Development Environment

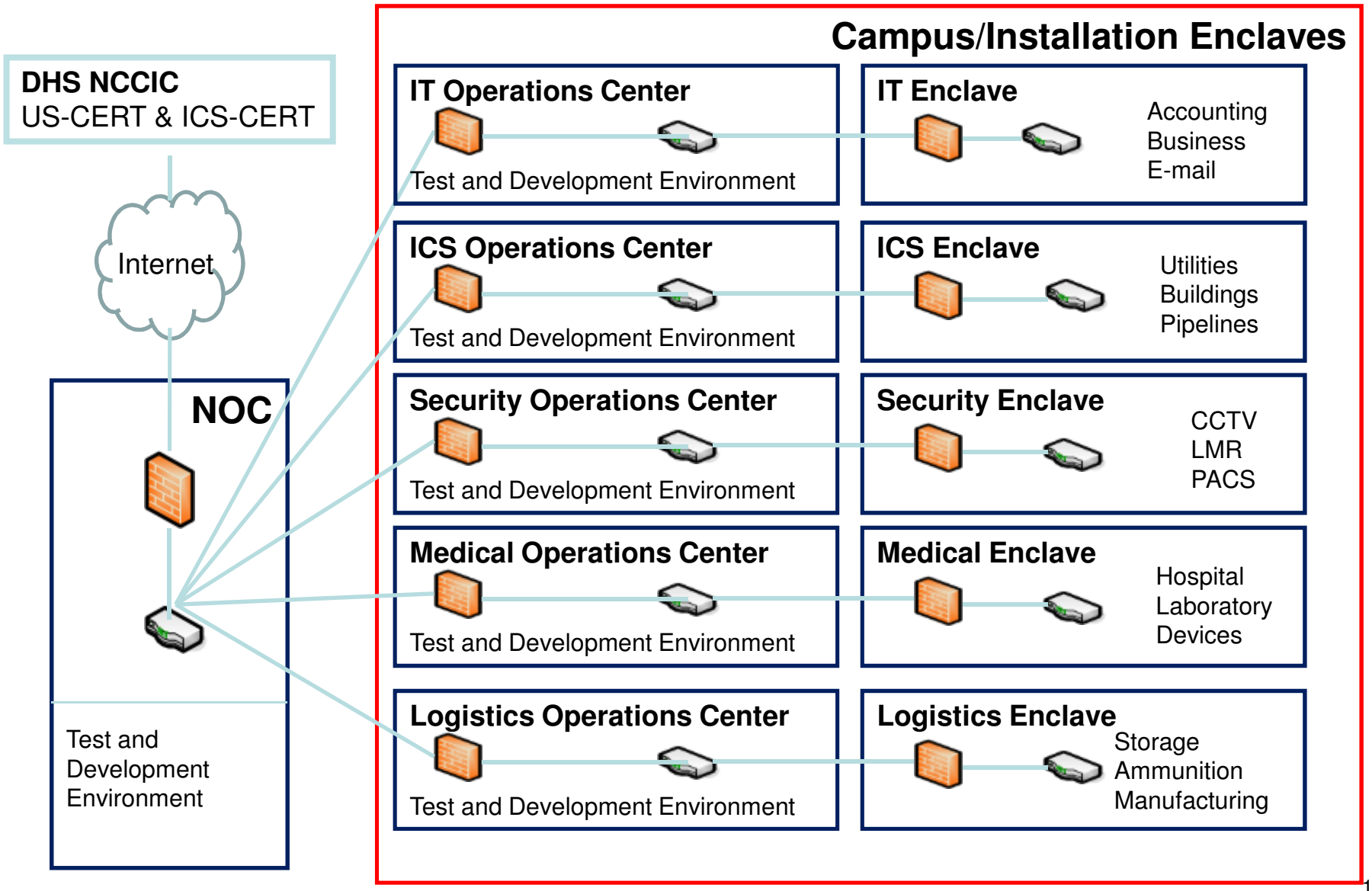
Step 1 – Inventory of the ICS Assets
- Sophia
- Nessus

Step 2 – Create ICS Enclaves Architecture Diagram
- CSET
- Visio
- SamuraiFTSU

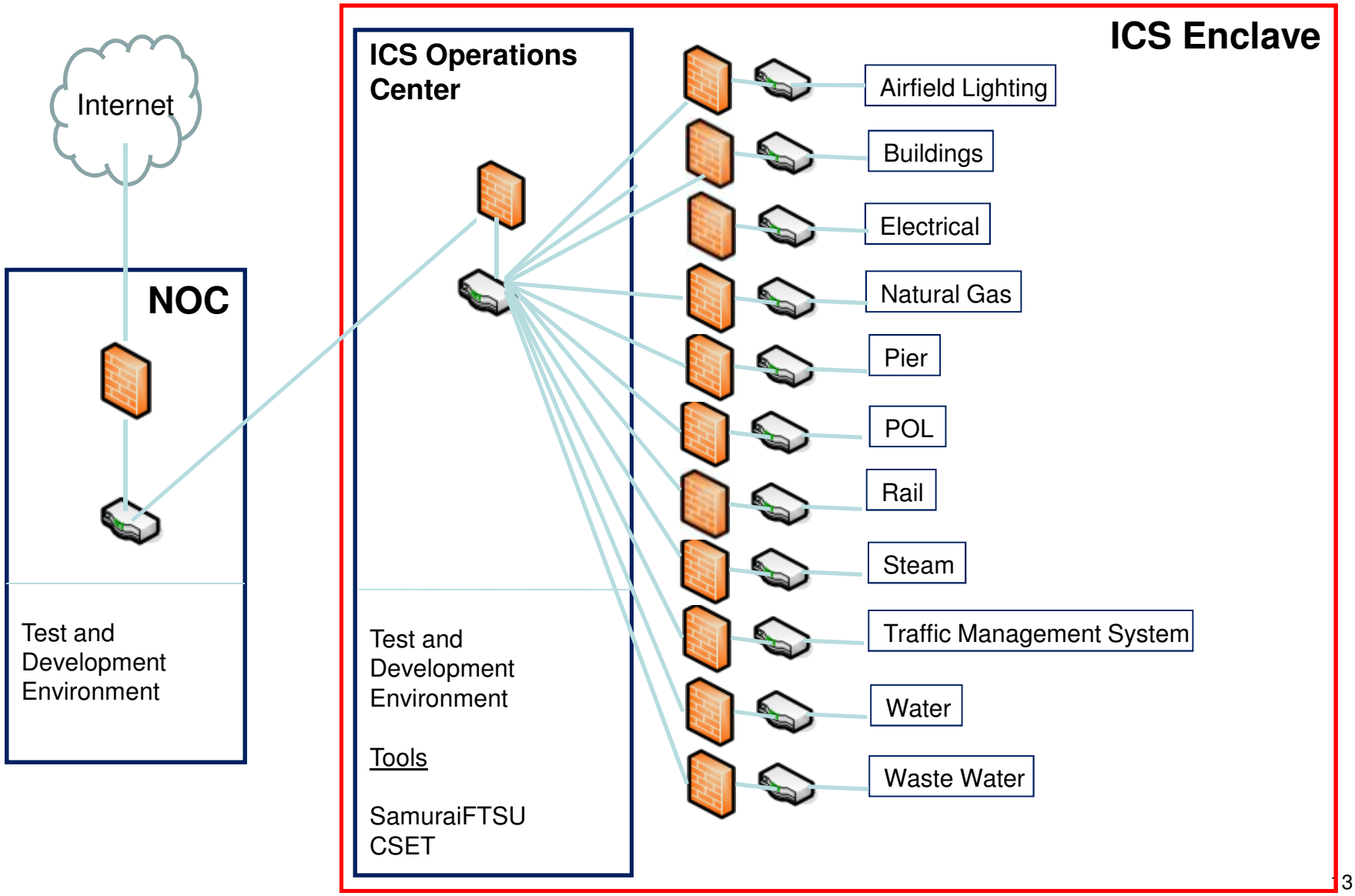
Step 3 – Perform Baseline Assessment
- CSET CNSSI 1253 ICS Overlay
- CSET NIST SP 800-82
- CSET NERC CIP

Step 4 – Conduct Penetration Testing of ICS System
- Kali Linux/Metasploit
- SamuraiFTSU

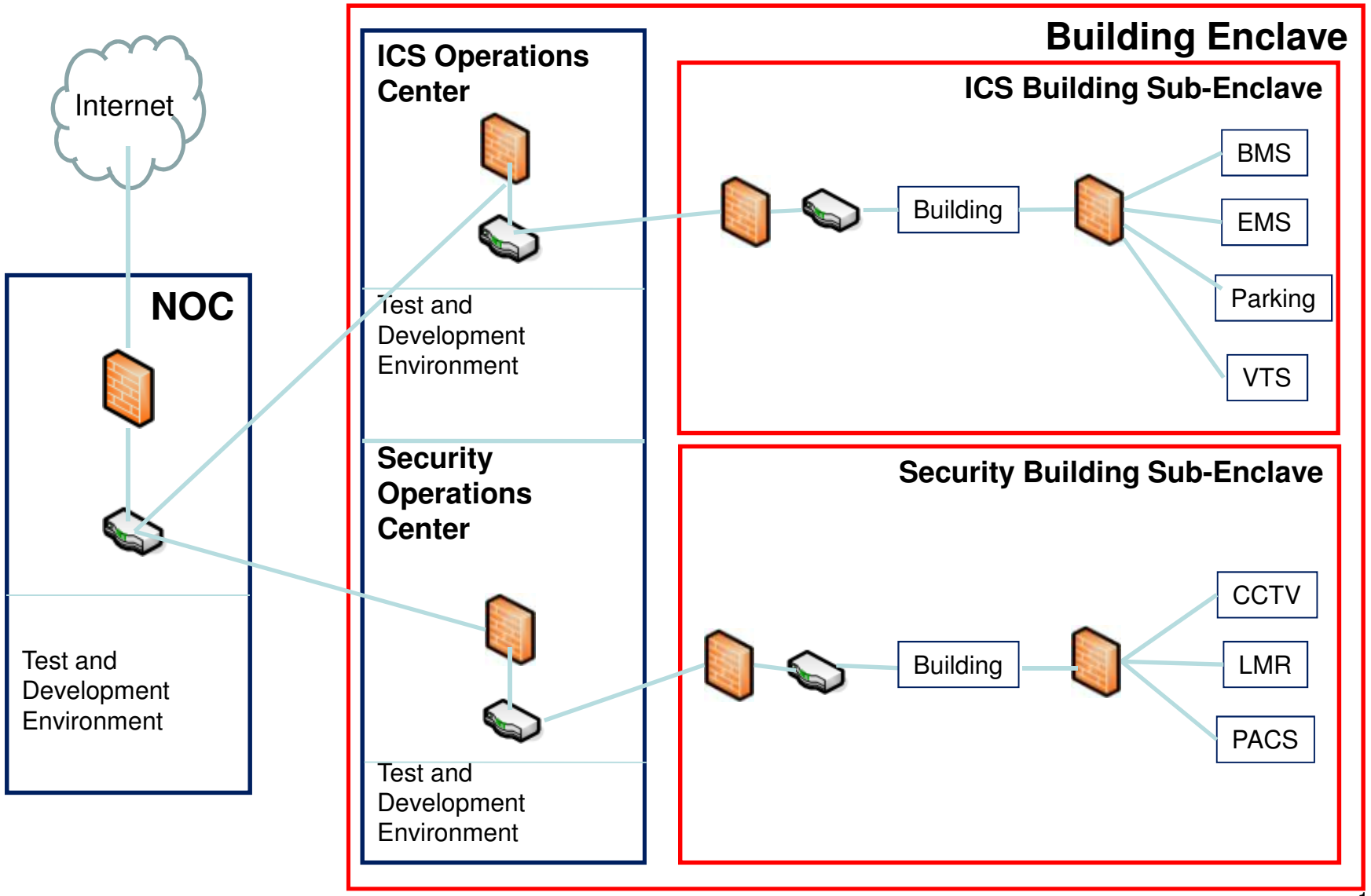
Campus/Installation Enclaves



ICS Enclave & Sub-Enclaves



ICS & Security Sub-Enclaves



ICS Test and Development Environment Tools

ICS Test and Development Environment Tools

- DHS CSET
- VMPlayer
- Kali Linux (VM with Metasploit)
- SamuariFTSU (VM with Wireshark)
- Metasploit (not in VM)
- Wireshark (not in VM)
- Snort
- Sophia

SamuraiFTSU Tool Suite

Network Pentest Tools
– nmap
– Nessus/NeXpose
– Metasploit
– Wireshark

Web Pentesting Tools
– Zed Attack Proxy
– Burp Suite
– w3af
– sqlmap
– BeEF

Wireless Pentest Tools
– Kismet
– Aircrack_ng
– KillerBee
– GNU Radio/GRC

HardwarePentest Tools
– GoodFET
– Bus Pirate
– TotalPhaseAardvark
– TotalPhaseBeagle
– entropy_graph
– bindiff

NESCOR Pentesting Process

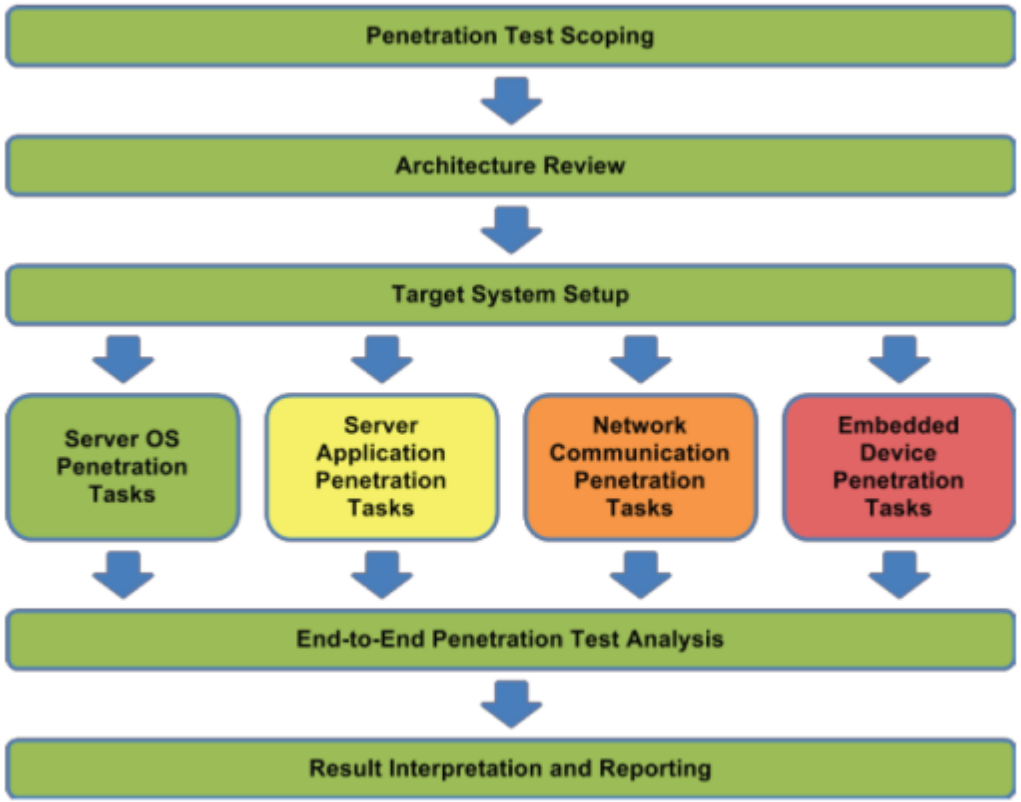


Figure 2a: Typical Penetration Testing Process

- Green: Tasks that should be performed most frequently, require the most basic of penetration testing skill, and can often be performed by internal security teams.
- Yellow: Tasks that are commonly performed and require moderate penetration testing skill.
- Orange: Tasks that are occasionally performed but may require higher levels of expertise.
- Red: Tasks that are infrequently performed and require highly specialized skills not often found in-house

NESCOR Common AMI Architecture

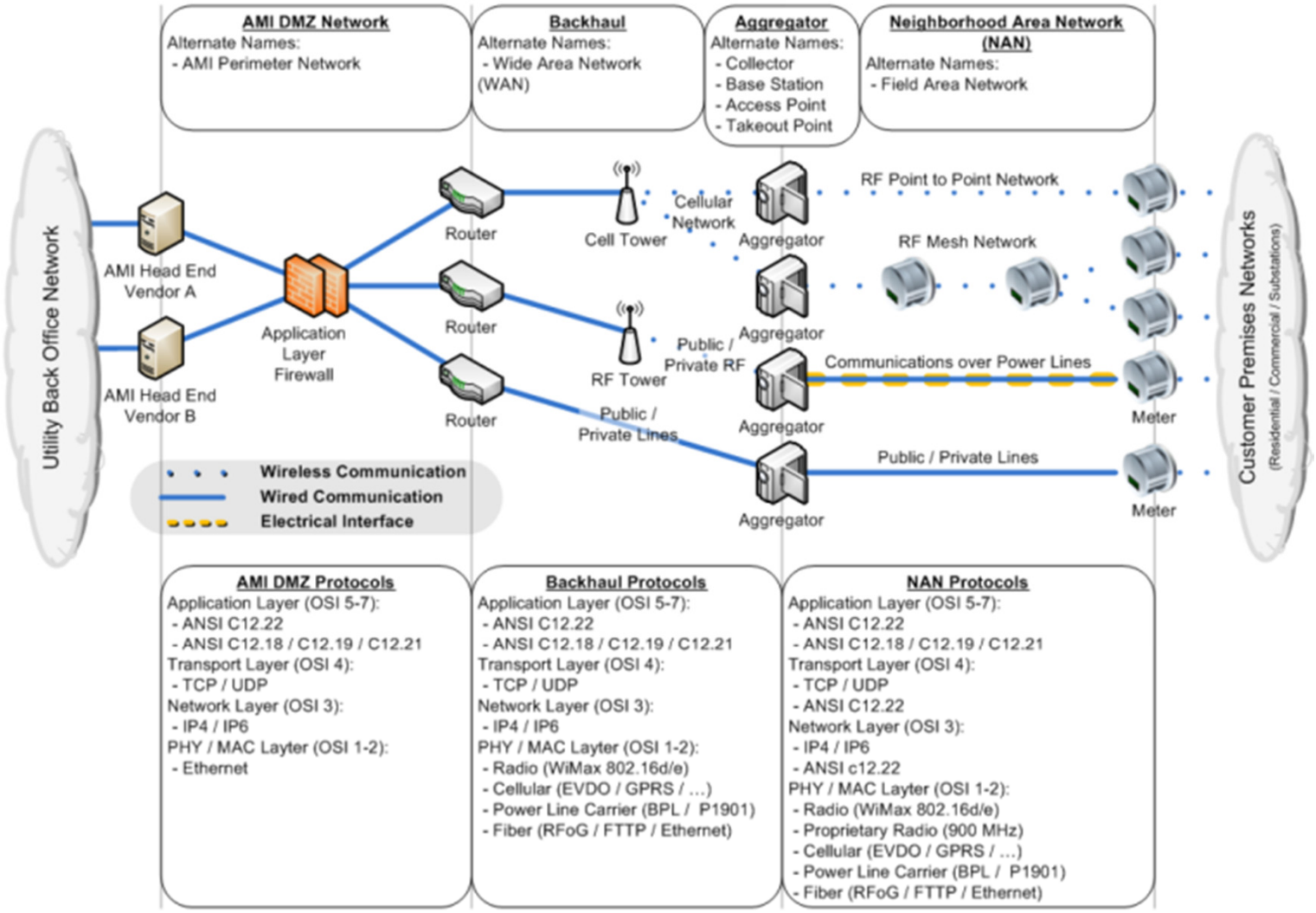


Figure 3.1a: Common AMI Architecture

NESCOR Typical DR Architecture

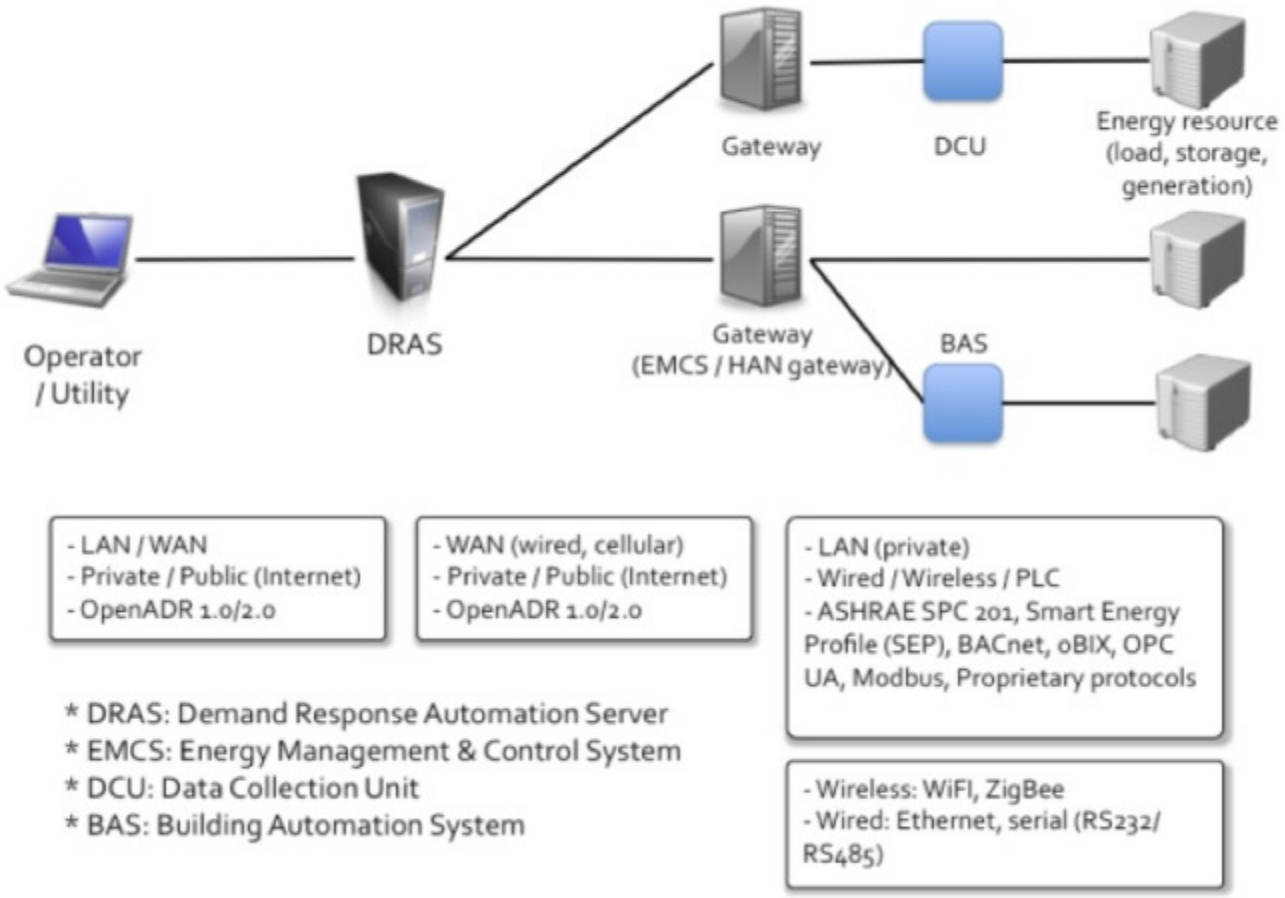


Figure 3.2a: Common DR Architecture

DR = Demand Response

NESCOR Typical DER Architecture

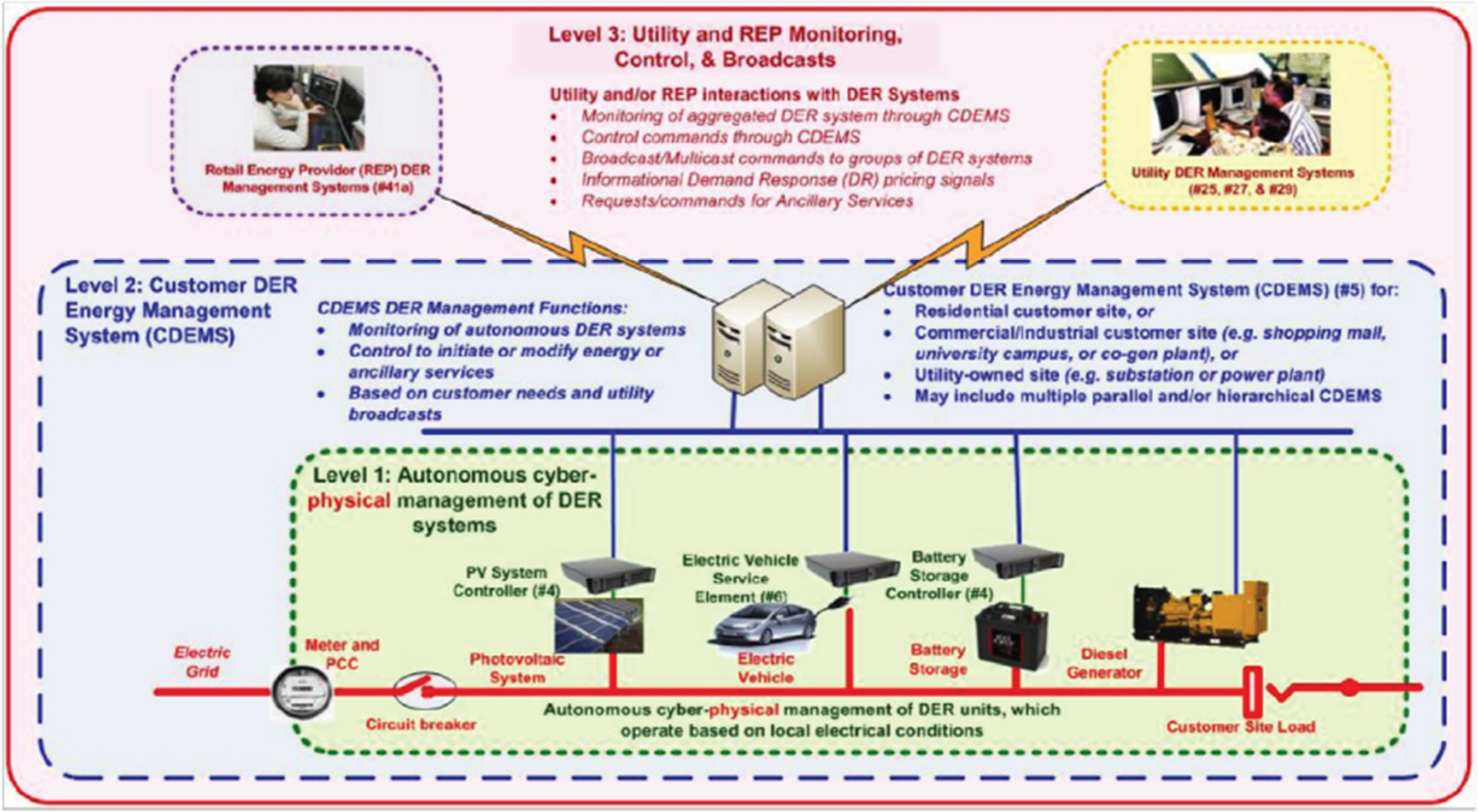


Figure 3.3a: Typical DER Architecture

DER = Distributed Energy Resources

NESCOR Typical AMI Dataflows

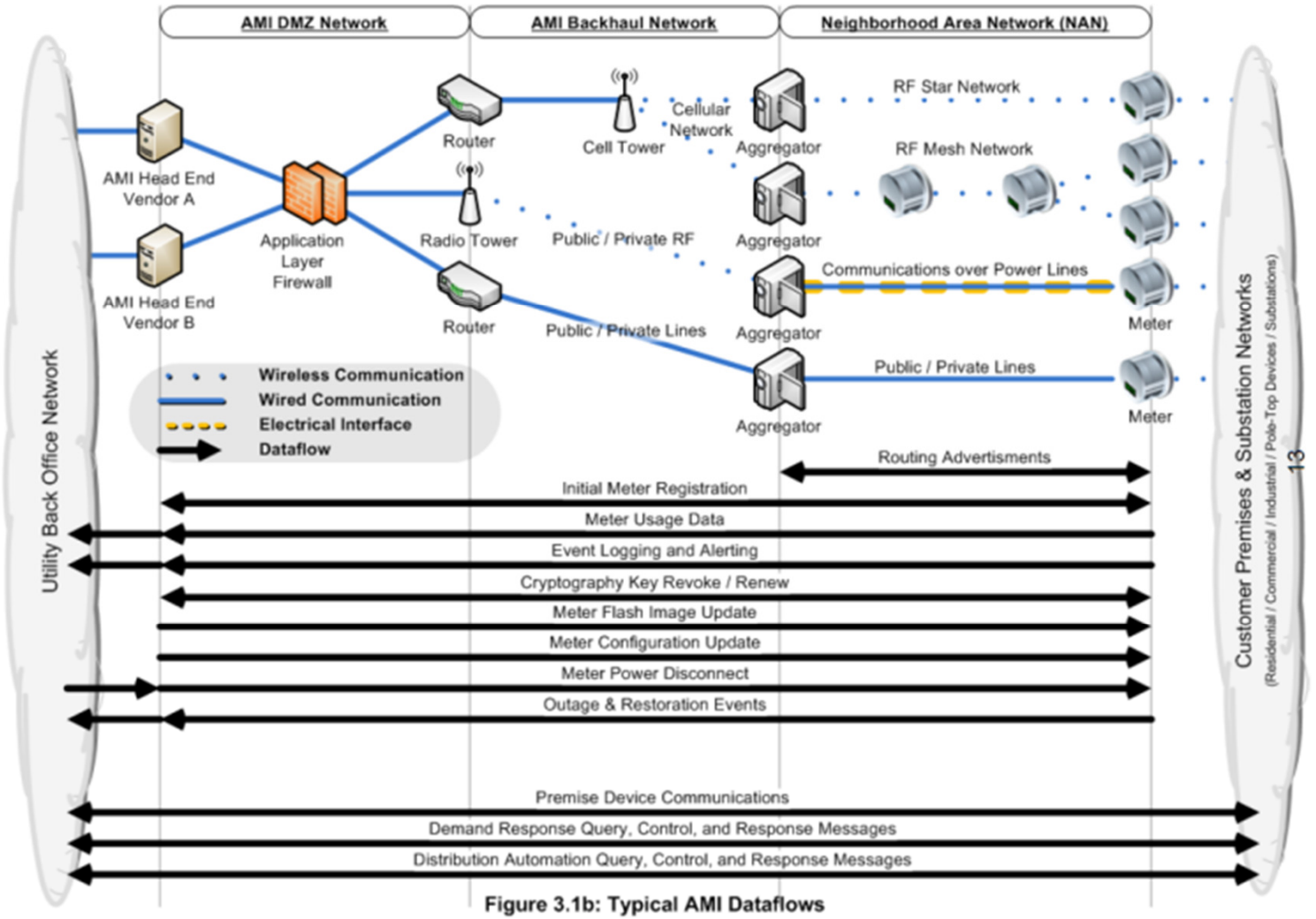


Figure 3.1b: Typical AMI Dataflows

NESCOR Common ET Architecture

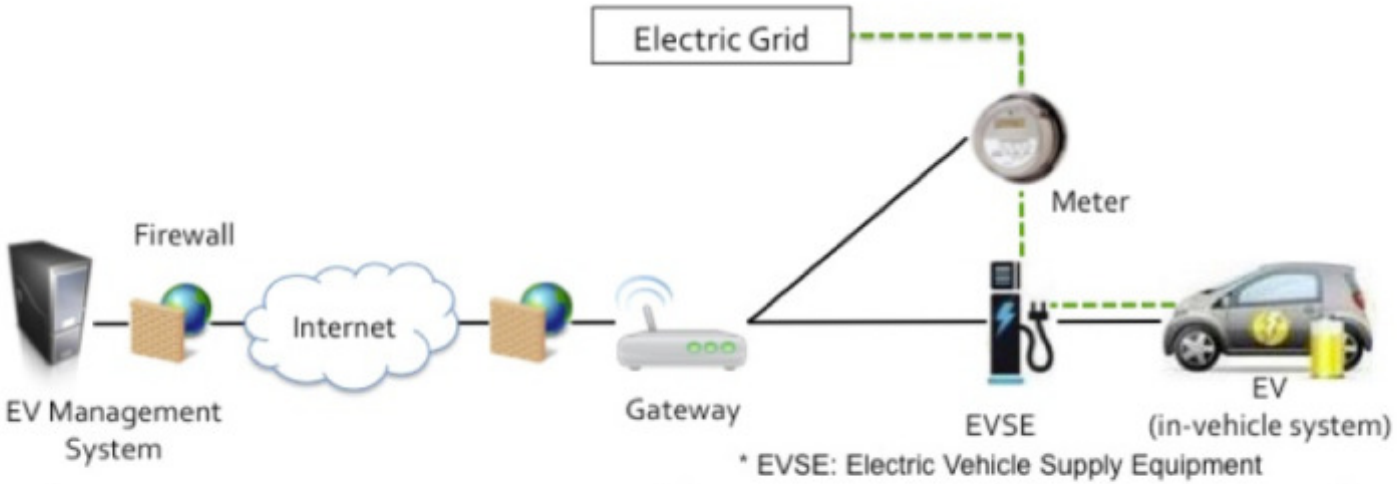


Figure 3.5a: Common ET Architecture

ET = Electric Transportation

NESCOR Embedded Device Pentest

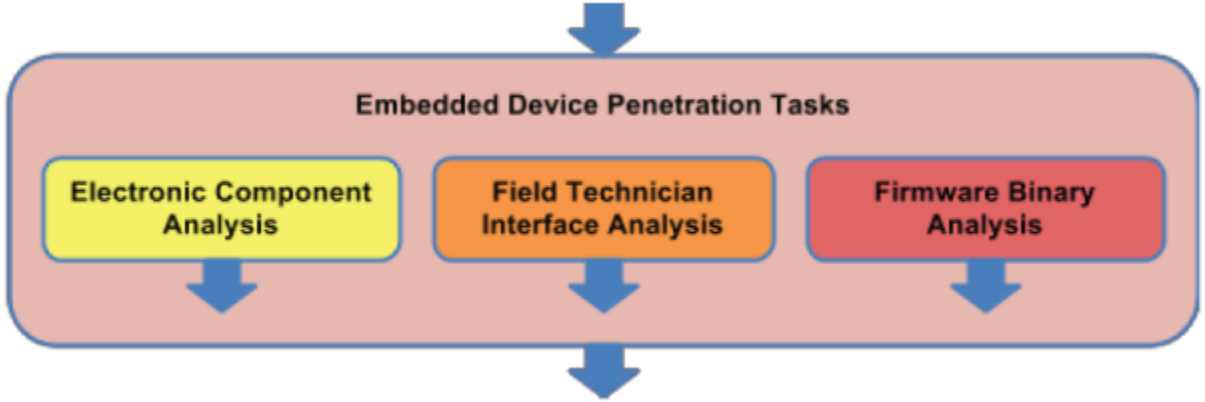


Figure 4a: Embedded Device Subcategory Flow

Suggested Tools:

- Basic tools such as screw drivers, wire cutters, pliers, tin snips, etc.
- Electronics equipment such as power supply, digital multimeter, and oscilloscope
- Electronic prototyping supplies such as breadboard, wires, components, alligator jumpers, etc.
- Specialized tools to communicate directly with individual chips or capture serial communications such as a Bus Pirate or commercial equivalent such as Total Phase Aardvark/Beagle.
- Universal JTAG tool such as a Bus Blaster, GoodFET, or a RIFF Box
- Surface mount micro test clips
- Electric meter test socket
- Disassembler Software for the appropriate microprocessors to be tested
- Entropy Analysis Software
- Protocol Analysis Software

NESCOR Electronic Component Analysis

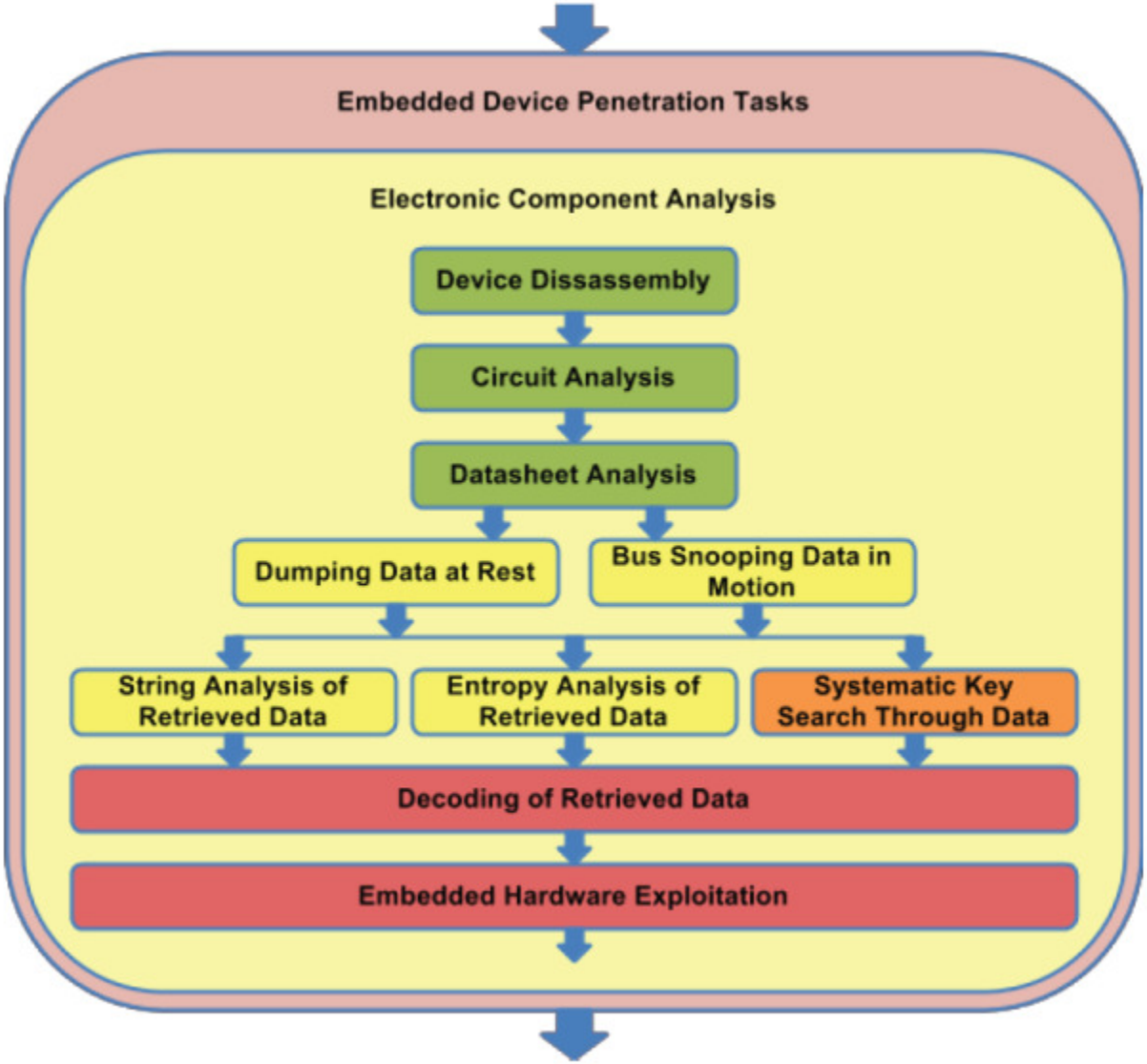


Figure 4.1a: Electronic Component Analysis Task Flow

NESCOR Field Technician Device Pentest

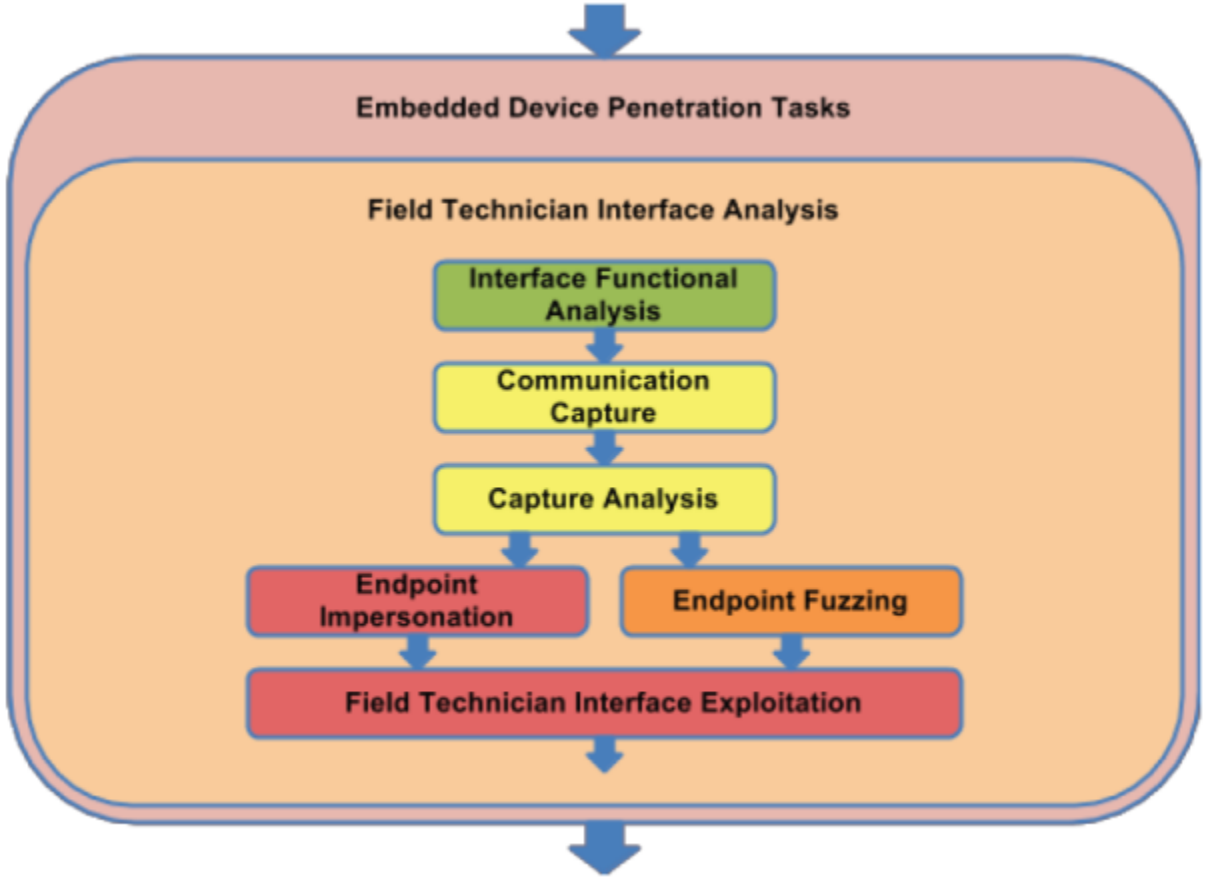


Figure 4.2a: Field Technician Device Task Flow

NESCOR Firmware Pentest

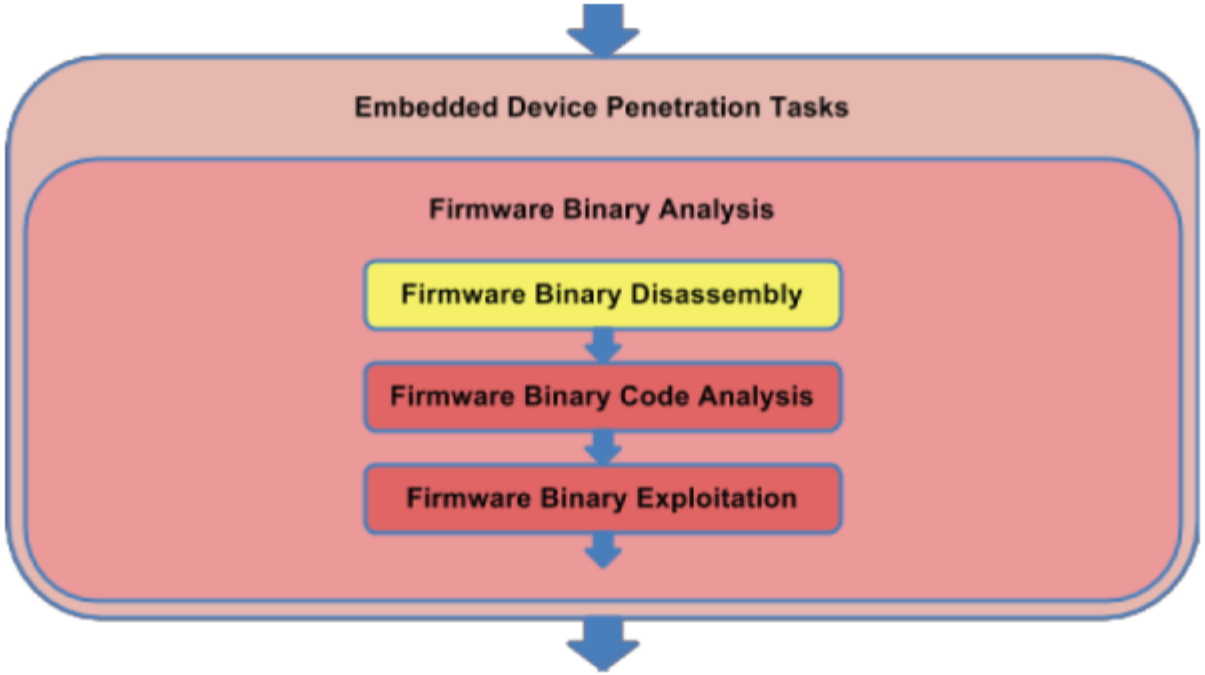


Figure 4.3a: Firmware Binary Analysis Task Flow

NESCOR Network Communications Pentest

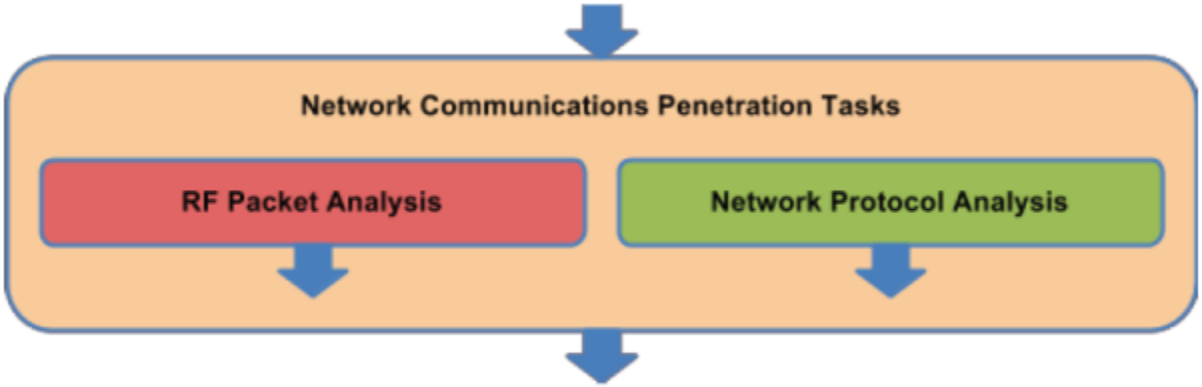


Figure 5a: Network Communications Subcategory Flow

Suggested Tools:

- Traffic capture and protocol decoder software such as Wireshark or tcpdump
- Hardware network taps
- Man-in-the-Middle tools such as Ettercap
- Protocol fuzzing tools such as Sulley
- Network packet generation such as Scapy
- Universal radio analysis kit, such as USRP2 with GNU Radio

NESCOR RF Packet Analysis Pentest

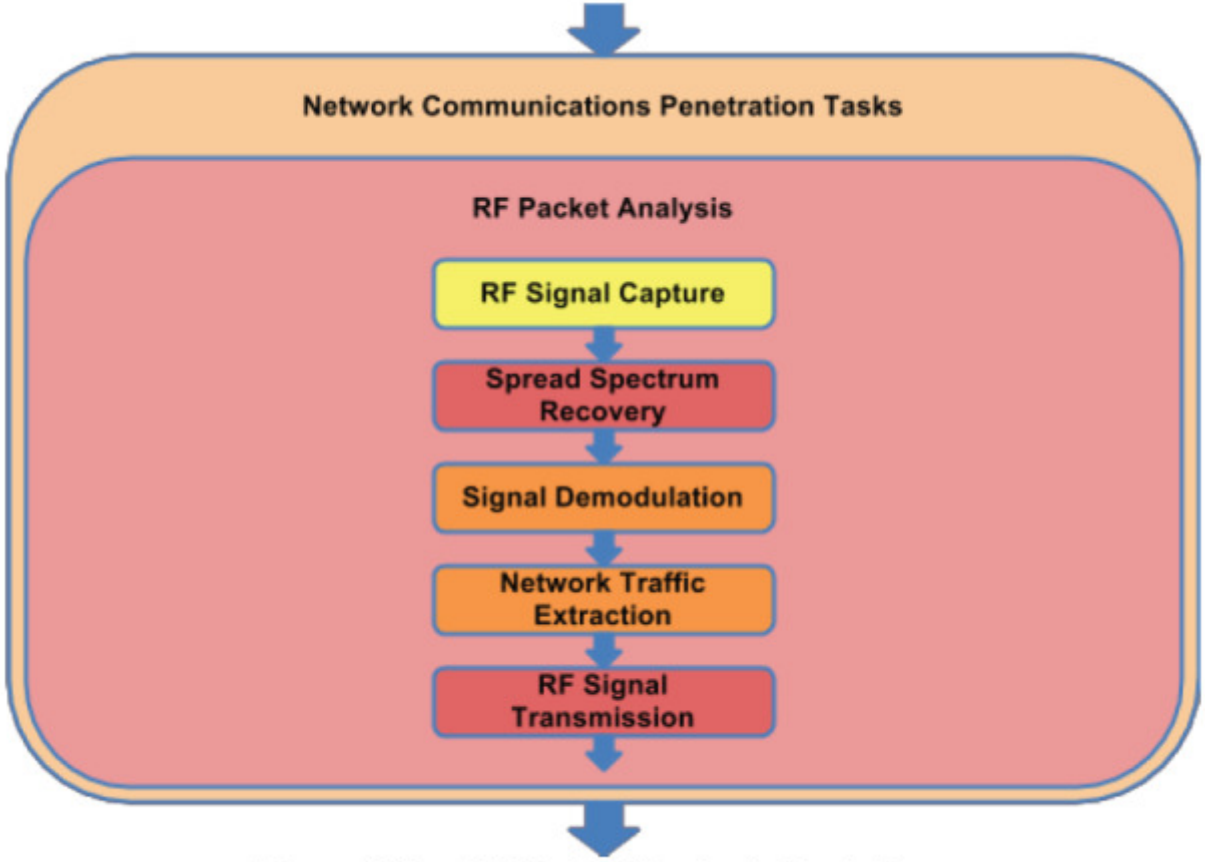


Figure 5.1a: RF Packet Analysis Task Flow

NESCOR Network Protocol Pentest

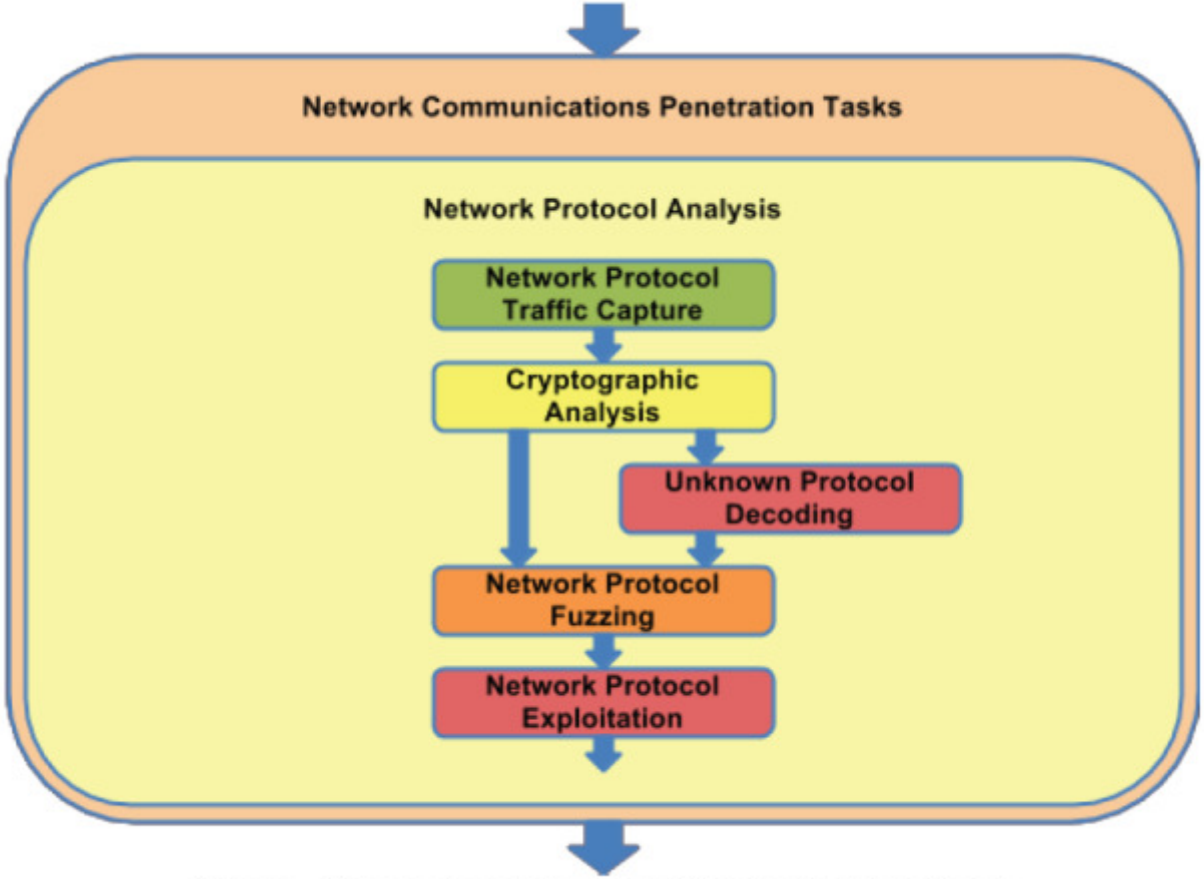


Figure 5.2a: Network Protocol Analysis Task Flow

ICS Test and Development Environment

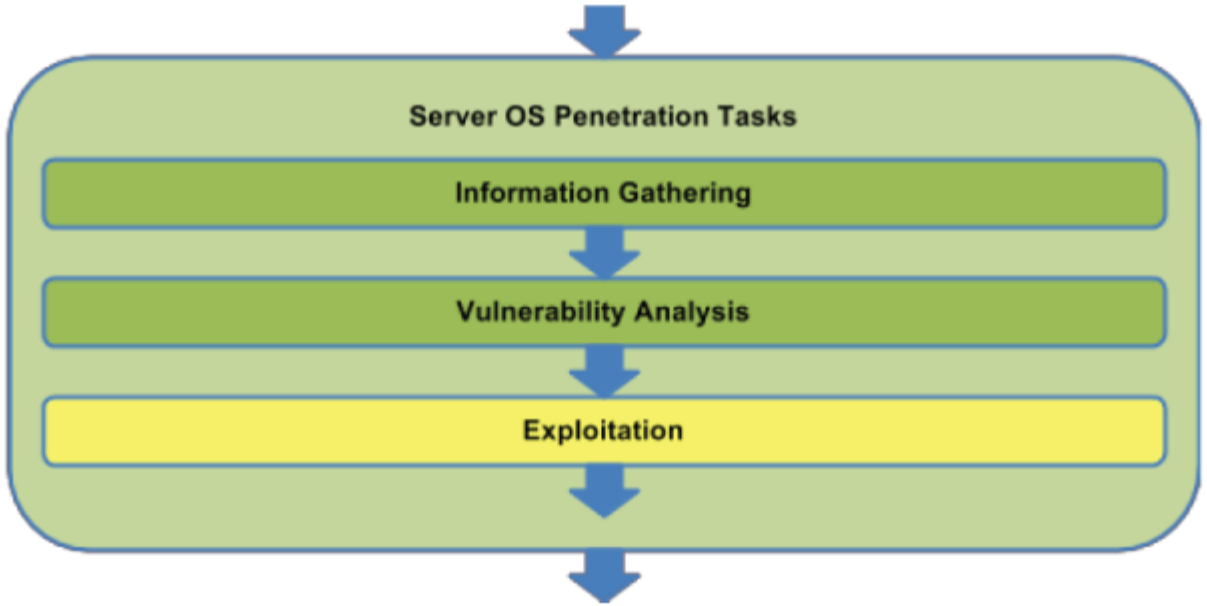


Figure 6a: Server OS Subcategory Flow

Suggested Tools:

- Standard network vulnerability assessment and penetration testing tools such as found on the BackTrack distribution
- *Guidance documents such as the Penetration Testing Standard (PTES)*

NESCOR OS Pentest

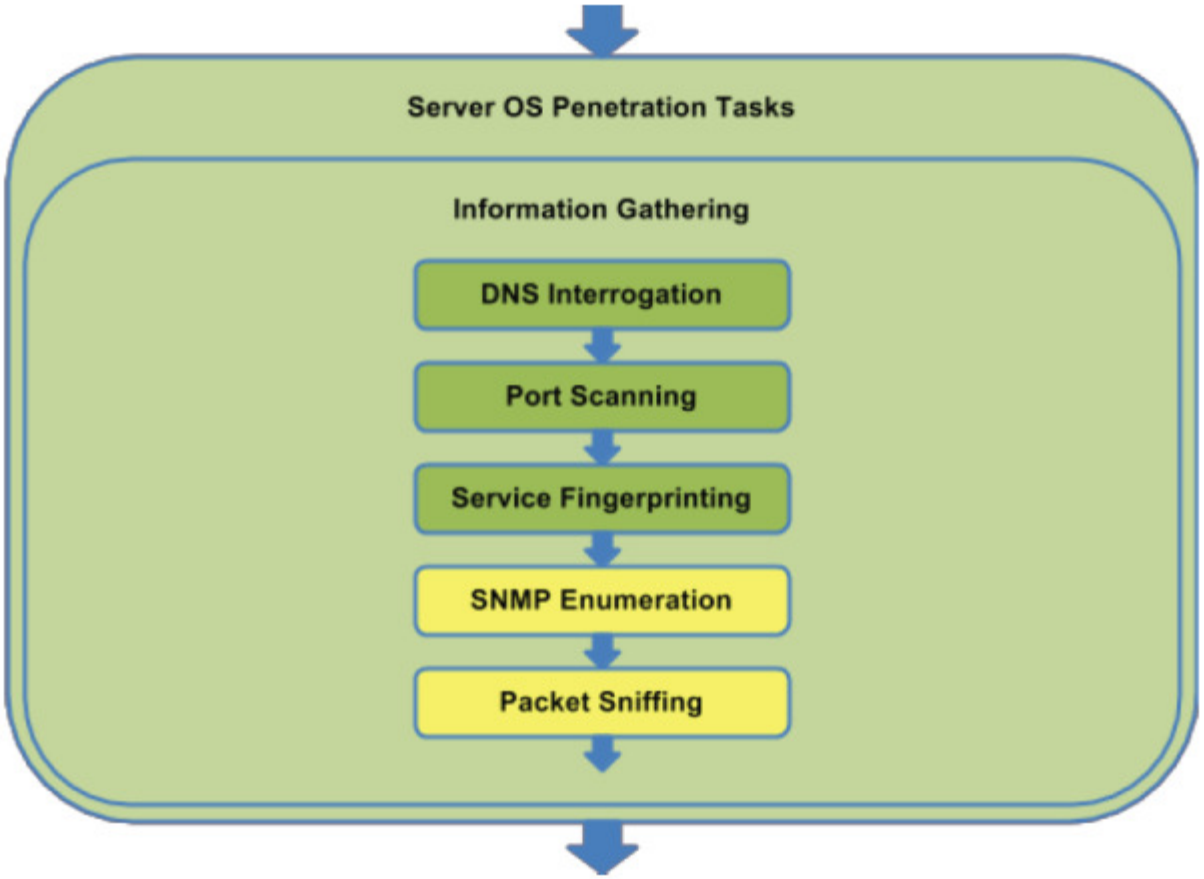


Figure 6.1a: OS Information Gathering Task Flow

NESCOR OS Vulnerability Pentest

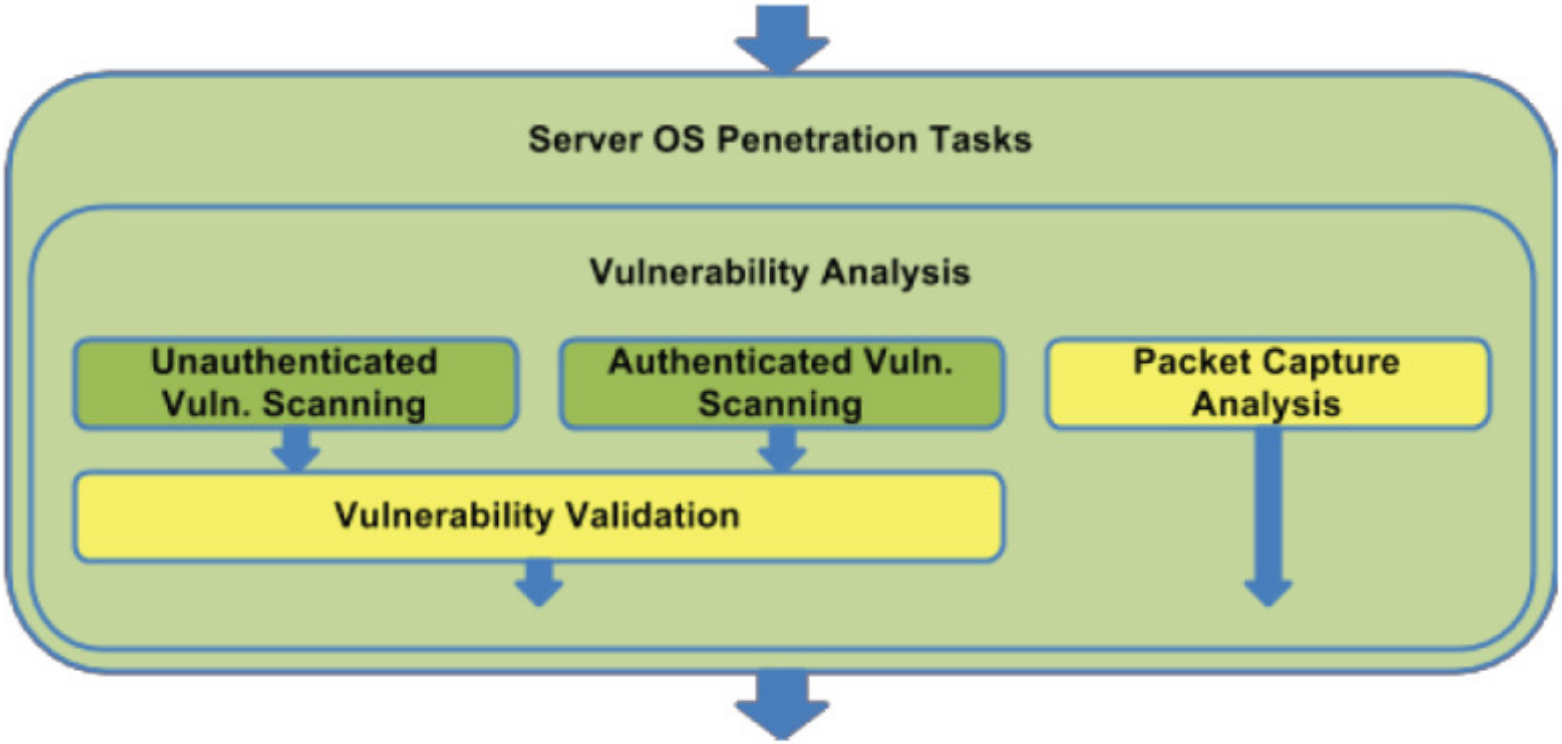


Figure 6.2a: OS Vulnerability Analysis Task Flow

NESCOR OS Exploitation Pentest

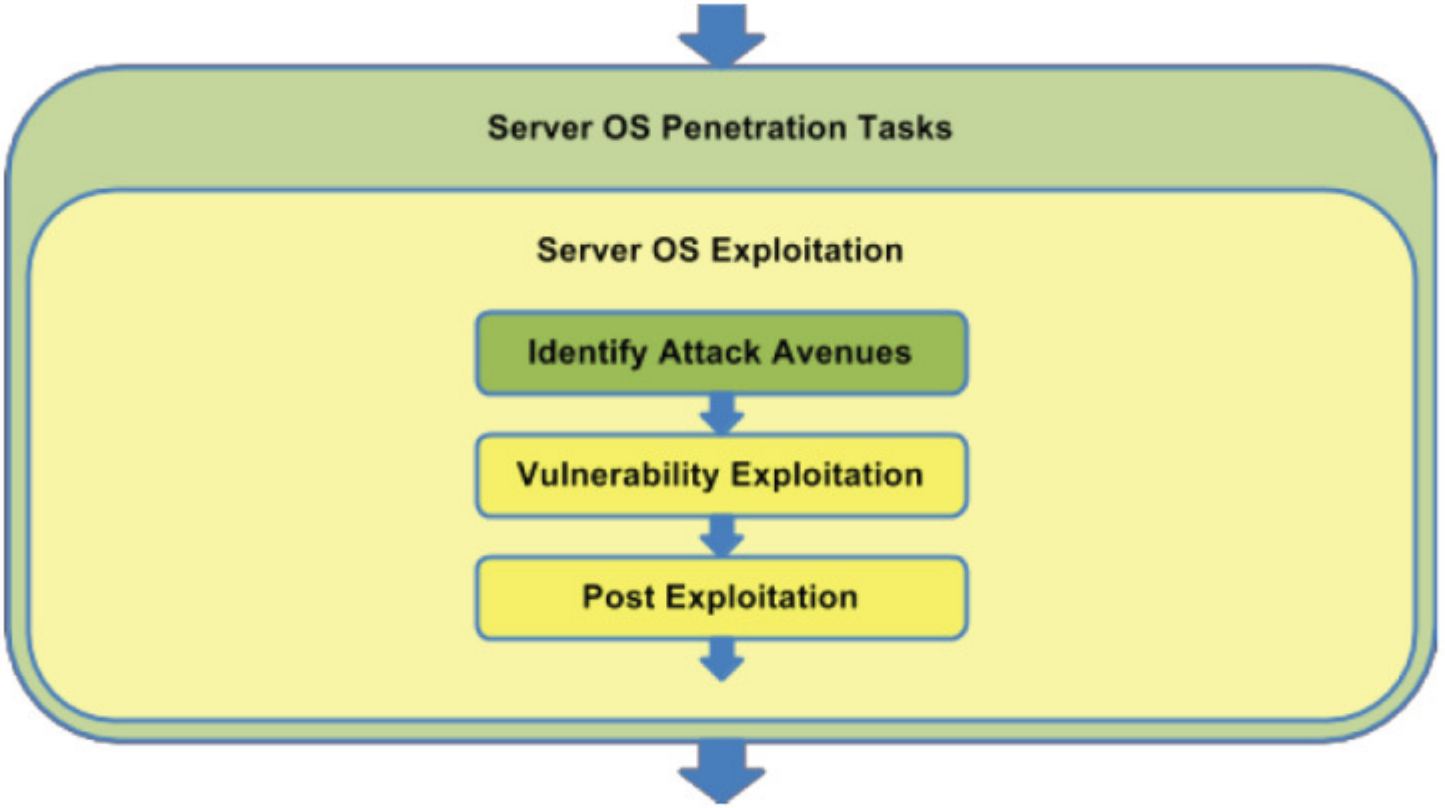


Figure 6.3a: Server OS Exploitation Task Flow

NESCOR Server Application Pentest

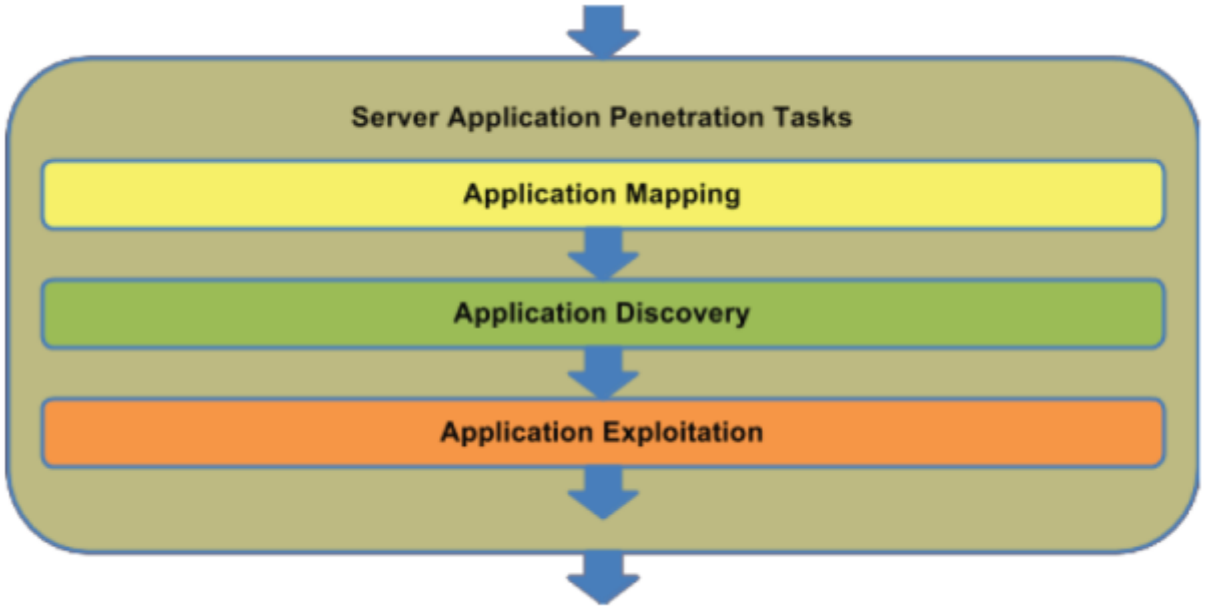


Figure 7a: Server Application Subcategory Flow

Suggested Tools:

- Web application penetration testing software such as found on the Samurai Web Testing Framework (SamuraiWTF) project

NESCOR Application Mapping Pentest

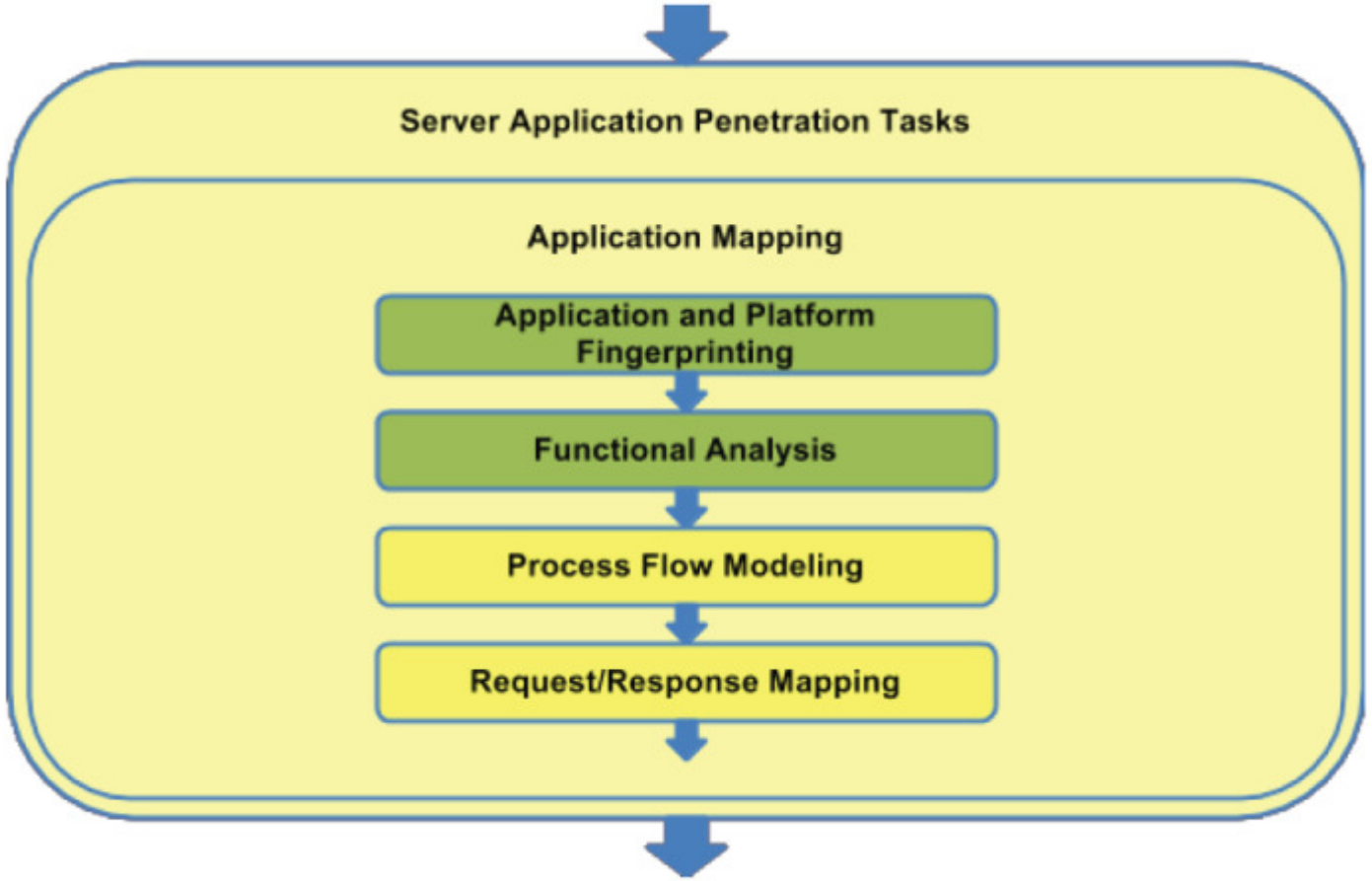


Figure 7.1a: Application Mapping Task Flow

ICS Test and Development Environment

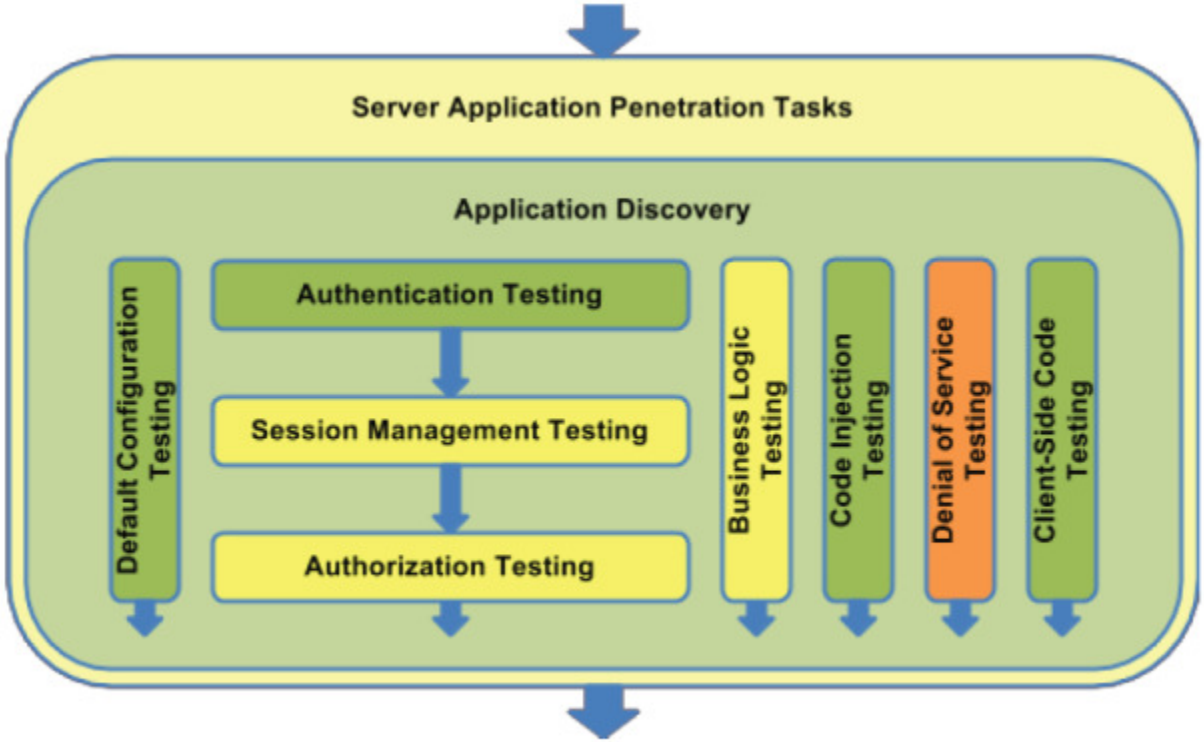


Figure 7.2a: Application Discovery Task Flow

NESCOR Server Exploitation Pentest

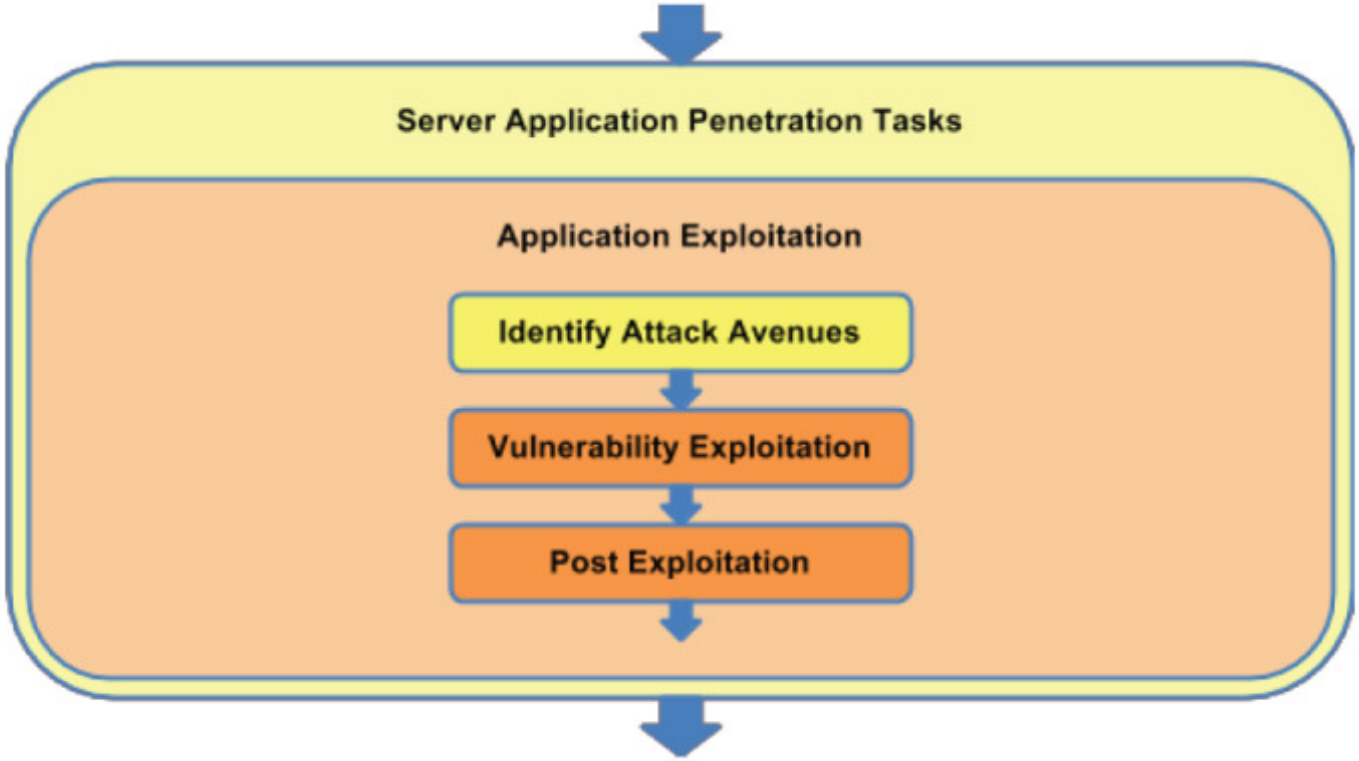


Figure 7.3a: Application Exploitation Task Flow