Network Security Systems Guide & RBAC Implementation

Prepared by: Baba Shaheer

Role-Based Access Control (RBAC) in Linux

Objective

Demonstrate RBAC by implementing three distinct roles: - **Admin**: Full access to system resources - **Developer**: Read & Write access to specific resources - **Auditor**: Read-only access to monitor resources

Implementation Steps

1. Creating Role-Based Groups

```
sudo groupadd admin
sudo groupadd developer
sudo groupadd auditor
```

2. Creating Users and Assigning Roles

```
# Create users and add them to their respective groups
sudo useradd -m -G admin adminuser
sudo useradd -m -G developer devuser
sudo useradd -m -G auditor audituser

# Set passwords for the users
echo "adminuser:AdminPass" | sudo chpasswd
echo "devuser:DevPass" | sudo chpasswd
echo "audituser:AuditPass" | sudo chpasswd
```

3. Creating a Secure Directory Structure

```
sudo mkdir /opt/securedata
sudo chown root:admin /opt/securedata
sudo chmod 770 /opt/securedata
```

4. Setting Access Permissions

```
# Give developer group read & write access
sudo setfacl -m g:developer:rw /opt/securedata

# Give auditor group read-only access
sudo setfacl -m g:auditor:r /opt/securedata
```

5. Verifying Role-Based Access Control

```
# Switch to admin user and create a file
su - adminuser
touch /opt/securedata/adminfile
exit

# Switch to developer user and try modifying the file
su - devuser
echo "Dev can write" >> /opt/securedata/adminfile  # Should work
exit

# Switch to auditor user and try modifying the file
su - audituser
```

```
echo "Audit attempt to write" >> /opt/securedata/adminfile  # Should fail
exit
```

Expected Behavior

- **Admin**: Can perform all operations (create/read/write/delete)
- **Developer**: Can read and modify files but with limited permissions
- **Auditor**: Can only read files, cannot make any modifications

---

Intrusion Detection and Prevention Systems

IDS (Intrusion Detection System)

An IDS monitors network traffic or system activities for suspicious behavior and alerts administrators when potential threats are detected.

Example: An IDS notices unusual login attempts occurring at 3 AM and sends an alert to the security team.

IPS (Intrusion Prevention System)

An IPS not only detects but also takes automatic action to block or prevent detected threats.

Example: When an IPS detects someone attempting a SQL injection attack on your web application, it automatically blocks that IP address.

NIDS (Network-based Intrusion Detection System)

A NIDS monitors network traffic across an entire network segment.

Example: Snort (open-source NIDS) monitoring all traffic passing through your company's internet gateway, looking for patterns that match known attack signatures.

NIPS (Network-based Intrusion Prevention System)

A NIPS monitors network traffic and can take immediate actions to prevent threats across the network.

Example: A NIPS detecting and blocking a DDoS attack before it overwhelms your web servers by identifying the abnormal traffic pattern.

HIDS (Host-based Intrusion Detection System)

A HIDS runs on individual hosts/devices to monitor activities occurring within that specific system.

Example: OSSEC monitoring file changes on a critical server and alerting when unauthorized modifications occur to system files.

HIPS (Host-based Intrusion Prevention System)

A HIPS monitors and protects a specific host/device and can take immediate action to block threats.

Example: McAfee Host IPS detecting an attempt to exploit a Windows vulnerability on a workstation and blocking the malicious process from executing.

Additional Security Components

Firewall

A barrier between a trusted network and an untrusted network that controls incoming and outgoing network traffic based on predetermined security rules.

Example: A firewall configured to allow only HTTP (port 80) and HTTPS (port 443) traffic to your web server, blocking all other connection attempts.

Honeypot

A decoy system designed to look like a legitimate part of the network but is actually isolated and monitored to attract and detect attackers.

Example: Setting up a fake server that appears to contain financial records but actually contains no sensitive data, allowing security teams to study attacker techniques when they break in.

Comparative Analysis

System	Function	Location	Response
IDS	Detection only	Network or Host	Passive (alerts)
IPS	Detection and Prevention	Network or Host	Active (blocks)
NIDS	Detection at network level	Network segments	Passive
NIPS	Prevention at network level	Network segments	Active
HIDS	Detection on single device	Individual hosts	Passive
HIPS	Prevention on single device	Individual hosts	Active
Firewall	Traffic filtering	Network perimeter	Active (blocks)
Honeypot	Attack analysis	Isolated environment	Passive (monitors)

Visualizing Security System Placement

```
                   INTERNET
                       |

                  [FIREWALL]
                       |

                    [NIPS]
                       |

      +----------------+----------------+
      |                                 |
[DMZ NETWORK]                    [INTERNAL NETWORK]
(Honeypot)                              |

                                  [HIPS/HIDS]
                                        |

                              [PROTECTED SYSTEMS]
```

Implementation Best Practices

For IDS/IPS

1. **Regular Updates**: Keep signature databases current

2. **Baseline Establishment**: Define normal network behavior
3. **Tuning**: Adjust sensitivity to reduce false positives
4. **Response Planning**: Document procedures for when alerts occur

For RBAC

1. **Principle of Least Privilege**: Grant only necessary permissions
2. **Role Separation**: Clearly define responsibilities
3. **Regular Auditing**: Review access rights periodically
4. **Documentation**: Maintain clear records of all role assignments

---

This document serves as a beginner-friendly introduction to network security systems and role-based access control implementation. For production environments, additional security measures should be implemented.