AppDynamics Machine Agent SSL Certificate Management Script

Functionality Analysis & Documentation

📋 Executive Summary

This bash script automates the process of configuring SSL certificate trust stores for AppDynamics Machine Agent installations. It discovers running agent processes, extracts SSL certificates from the AppDynamics controller, creates a custom truststore, and reconfigures the agent to use the new certificates.

🎯 Primary Objectives

Core Purpose

- **Automate SSL certificate management** for AppDynamics Machine Agent
- **Resolve SSL trust issues** between Machine Agent and Controller
- **Eliminate manual certificate configuration** steps
- **Ensure secure communication** with AppDynamics SaaS controllers

Key Benefits

- ✅ **Zero-downtime certificate updates**
- ✅ **Automatic controller discovery**
- ✅ **Self-contained execution** (no external dependencies)
- ✅ **Immediate service restart** and validation

🔧 Detailed Functionality Breakdown

Phase 1: Discovery & Validation

```bash
# Process Discovery
PID=$(pgrep -f '[m]achineagent.jar' || true)
```

Actions Performed:

1. **Machine Agent Process Detection**

   - Searches for running `machineagent.jar` processes
   - Uses process grep with regex pattern matching
   - Validates agent is currently running
   - **Error Handling:** Exits if no agent process found

2. **Machine Agent Installation Path Discovery**
   - Extracts JAR path from process command line
   - Resolves symbolic links to actual file paths
   - Determines `MACHINE_AGENT_HOME` directory
   - **Output:** Displays discovered installation path

3. **Controller Configuration Extraction**
   - Locates `controller-info.xml` configuration file
   - Parses XML to extract controller hostname
   - Uses grep with Perl-compatible regex for precision
   - **Validation:** Ensures controller host is successfully extracted

---

Phase 2: Certificate Management

2.1 Environment Preparation

```bash
rm -f *.pem *.jks
```

Actions Performed:

- **Cleanup existing certificates** (PEM and JKS files)
- **Prepare clean workspace** for new certificate operations
- **Prevent conflicts** with previous certificate files

2.2 SSL Certificate Discovery

```bash
"$JRE_BIN_KEYTOOL" -printcert -sslserver "$controllerName"
```

Actions Performed:

- **Connect to AppDynamics controller** via SSL
- **Retrieve complete certificate chain** from server
- **Display certificate information** for verification
- **Validate SSL connectivity** to controller

2.3 Certificate Extraction & Processing

```bash
"$JRE_BIN_KEYTOOL" -printcert -sslserver "$controllerName" -rfc > controllercerts.pem
csplit -z controllercerts.pem /-----BEGIN/ '{*}' --prefix='cert'
```

Actions Performed:

1. **Download certificate chain** in RFC format (PEM)
2. **Split combined PEM file** into individual certificates
3. **Create separate certificate files** for each cert in chain
4. **Prepare certificates** for truststore import

Phase 3: Truststore Creation

3.1 Certificate Import Process

```bash
for inputfile in cert*; do
  "$JRE_BIN_KEYTOOL" -import -noprompt -alias "$inputfile" -file "${inputfile}.pem" -keystore truststore.jks -storepass c
done
```

Actions Performed:

- **Iterate through all certificate files**
- **Import each certificate** into new truststore
- **Use unique aliases** for each certificate
- **Set standard truststore password** (`changeit`)
- **Create Java KeyStore (JKS)** format truststore

3.2 Truststore Deployment

bash

```
mv truststore.jks "$MACHINE_AGENT_HOME/conf/"
```

Actions Performed:

- **Deploy truststore** to agent configuration directory
- **Ensure proper file location** for agent access
- **Verify truststore contents** using keytool list command

Phase 4: Agent Configuration

4.1 Startup Script Modification

bash

```
TRUSTSTORE_FLAG="-Djavax.net.ssl.trustStore=\${MACHINE_AGENT_HOME}/conf/truststore.jks -Djavax.net.ssl.trustSto
sed -i "s|\(JAVA_OPTS.*-Xmx256m\)|\1 $TRUSTSTORE_FLAG|" "$AGENT_SCRIPT"
```

Actions Performed:

- **Locate machine-agent startup script**
- **Check for existing truststore configuration**
- **Inject SSL truststore parameters** into JAVA_OPTS
- **Preserve existing JVM arguments**
- **Avoid duplicate configuration** entries

4.2 Service Restart & Validation

bash

```
systemctl stop appdynamics-machine-agent.service
systemctl start appdynamics-machine-agent.service
```

Actions Performed:

- **Gracefully stop** Machine Agent service
- **Wait for clean shutdown**
- **Start service** with new configuration

- **Ensure service activation** with updated truststore

🔍 Technical Implementation Details

Security Considerations

Aspect	**Implementation**	**Security Level**
Certificate Validation	Direct SSL server connection	✅ High
Truststore Password	Standard Java default (`changeit`)	⚠️ Medium
File Permissions	Inherits system defaults	⚠️ Medium
Process Discovery	Pattern-based process matching	✅ High

Error Handling Mechanisms

1. **Process Validation:** Script exits if Machine Agent not running
2. **File Validation:** Checks for required configuration files
3. **Tool Validation:** Verifies keytool availability
4. **Connection Validation:** Tests SSL connectivity to controller

Dependencies & Requirements

- **Operating System:** Linux with systemd
- **Java Runtime:** Bundled with Machine Agent
- **System Tools:** `pgrep`, `grep`, `sed`, `csplit`
- **Permissions:** Root access for service management
- **Network:** SSL connectivity to AppDynamics controller

🚀 Execution Flow Summary

```
mermaid
```

```
graph TD
    A[Script Start] --> B[Find Machine Agent PID]
    B --> C[Locate Agent Installation]
    C --> D[Extract Controller Info]
    D --> E[Clean Existing Certificates]
    E --> F[Download Controller Certificates]
    F --> G[Split Certificate Chain]
    G --> H[Create Truststore]
    H --> I[Deploy Truststore]
    I --> J[Update Agent Script]
    J --> K[Restart Agent Service]
    K --> L[Validation Complete]
```

Execution Time

- **Typical Duration:** 30-60 seconds
- **Network Dependent:** SSL certificate download
- **Service Restart:** ~10-15 seconds

⚠️ Prerequisites & Considerations

System Requirements

- ✅ **Linux system** with systemd service management
- ✅ **Running AppDynamics Machine Agent**
- ✅ **Root/sudo privileges** for service control
- ✅ **Network connectivity** to AppDynamics controller
- ✅ **Java keytool** (bundled with agent)

Operational Impact

- **Service Downtime:** Brief interruption during restart (~15 seconds)
- **Configuration Changes:** Permanent modification to startup script
- **File System Changes:** New truststore files in conf directory
- **Network Activity:** SSL certificate downloads from controller

Potential Risks

- ⚠️ **Service Interruption:** Brief monitoring gap during restart

- ⚠️ **Configuration Overwrite:** Existing truststore configurations replaced
- ⚠️ **Network Dependencies:** Requires controller accessibility

---

🎯 Use Cases & Scenarios

Primary Use Cases

1. **Initial SSL Setup:** First-time certificate configuration
2. **Certificate Renewal:** Updating expired or changed certificates
3. **Controller Migration:** Switching to new controller endpoints
4. **SSL Troubleshooting:** Resolving certificate trust issues

Ideal Deployment Scenarios

- **AppDynamics SaaS environments** with custom SSL certificates
- **On-premises controllers** with self-signed certificates
- **Enterprise environments** with corporate certificate authorities
- **Automated deployment pipelines** requiring SSL configuration

---

📊 Success Criteria & Validation

Script Success Indicators

- ✅ Machine Agent process discovered successfully
- ✅ Controller certificates downloaded and imported
- ✅ Truststore created and deployed
- ✅ Agent service restarted without errors
- ✅ SSL connectivity established to controller

Post-Execution Verification

```bash
```

```
# Verify truststore contents
keytool -list -keystore conf/truststore.jks -storepass changeit

# Check agent service status
systemctl status appdynamics-machine-agent.service

# Monitor agent logs for SSL errors
tail -f logs/machine-agent.log
```

🔧 Maintenance & Troubleshooting

Common Issues & Solutions

Issue	Cause	Solution
Agent not found	Service not running	Start Machine Agent service
Certificate download fails	Network/firewall issues	Check controller connectivity
Service restart fails	Permission issues	Run with sudo/root privileges
Truststore creation fails	Disk space/permissions	Check filesystem permissions

Monitoring Recommendations

- **Schedule regular execution** for certificate updates
- **Monitor agent logs** for SSL-related errors
- **Verify controller connectivity** before execution
- **Test in non-production** environments first

This analysis provides a comprehensive overview of the AppDynamics Machine Agent SSL certificate management script functionality, implementation details, and operational considerations.