Lessons Learned in Designing SE-Linux into A Toolkit for Secure Electric Power Grid Control Systems

Stanley A. Klein

Stan Klein Associates, LLC and

Open Secure Energy Control Systems, LLC

(301) 881-4087

sklein@cpcug.org

This work was supported by the Homeland Security Advanced Research Projects Agency (HSARPA)

Agenda

- Overview of project
- Overview of electric power SCADA
- SCADA security issues
- Overview of IEC-61850
- Toolkit features, security, and potential applications
- Use of SE-Linux in the toolkit
- Challenges to future SE-Linux development

Project Overview

- Phase II Small Business Innovation Research Project
- Responded to DHS SBIR topic on SCADA security
- Project focus:
  - Protection of IEC-61850-based systems
  - Secure infrastructrure and applications
  - Overall SCADA/Control Center functionality
  - Open source software

Electric Power SCADA Technology Overview

- Monitor and control substation devices from a center
- SCADA networks are usually dedicated
- Control centers are SCADAs enhanced with "what if," optimization, and other advanced applications
- Other communications requirements
    - With other utility control centers
    - Between substations
    - Corporate and power market

SCADA Security Overview

- Most existing substation equipment has minimal security protection
- Most security upgrades must be external to devices
- Focus for security upgrades is Role Based Access Control
- Individual utilities must be able to
  - Define the roles and business process rules appropriate to their power systems
  - Have utility personnel manage the roles and rules
- Rules may need to change with threat and system conditions

SCADA Security: Example roles

- Power system operator
- Power system operations supervisor
- Protective relay engineer
- Protective relay engineering supervisor
- Substation equipment maintainer
- Corporate data user
- Non-personally-specific roles for particular tasks, e.g., two roles for the task, anyone can perform either, one person can't perform both

SCADA Security: Example rules

- Relay settings only by protective relay engineers
- Relay settings require business process reviews and approvals
- Predefined relay setting groups can be selected by system operators
- Relay settings and status can be viewed by operators
- Changes to safety tags, data setting changes, and control commands for safety tagged equipment require supervisory approval

SCADA Security: Access control issues

- Many access control issues are embedded in application business logic and message content
- Access controlled objects are comparable to individual database records and fields
- Users are:
  - Known to the OS on their own workstation
  - Known to the application on control center servers and substation devices across the network
  - Not known to the OS on servers and devices
- Access control requirements often similar to firewalls

Overview of IEC-61850

- Object models replace numbered points
- Self-discovery simplifies system management
- Communication to control center uses Manufacturing Messaging Specification over ISO OSI protocol stack with TCP/IP transport via RFC 1006
- Direct LAN messaging between substation devices
- Object models are organized into base classes, common classes, and device logical nodes
- Object models are translatable to XML

61850 Base Class and Service Examples

- Base class examples
  - Logical Device
  - Logical Node
  - Data
  - Data attribute
  - Data set
  - Setting group
  - Buffered report control block
  - Log control block
  - Control

- Service examples
  - Get (e.g., data values)
  - Set (e.g., control block values)
  - Select setting group
  - Define data set
  - Control select
  - Control operate

Security Benefits of 61850

- Access control easier with named data items
- More alternatives for encryption and authentication
- Use of XML for configuration simplifies management
- XML simplifies imposing security controls on data objects
- Use of TCP/IP enables conventional network firewalls
- Object model accommodates security violation reporting

Toolkit features

- Polling/scheduling, alarm, and other SCADA functions
- Native support for 61850 object models and services
- MMS protocol over TCP/IP for substation communications
- XML for configuration, HMI screen definition, and management
- XML/SOAP/WSDL for internal & external messaging
- SSL/TLS or IPSEC network encryption
- Advanced applications and external interfaces

Toolkit Security Features

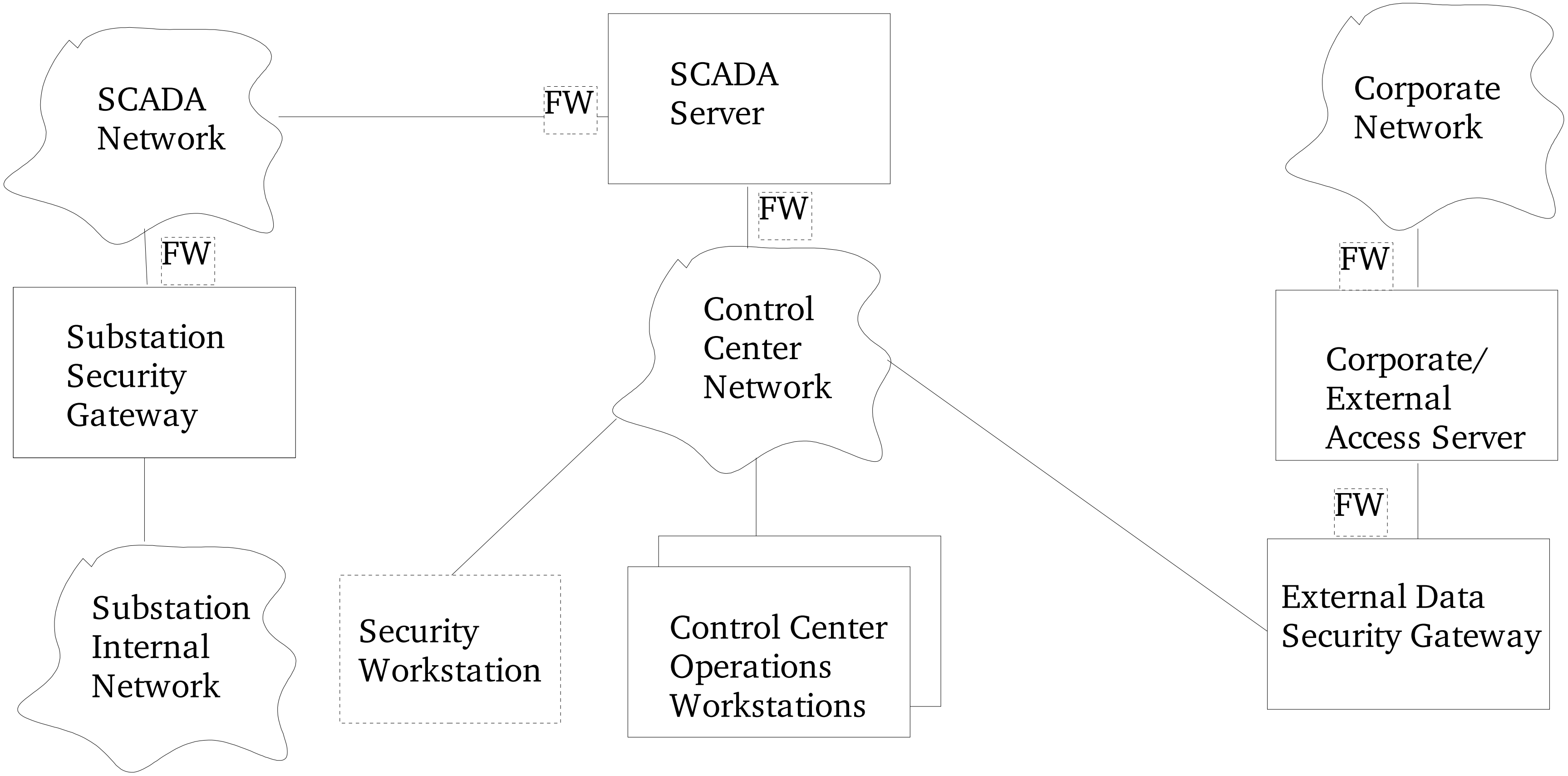
- SE-Linux supports mandatory access control for platform protection
- Application access control focused on messages
- Network encryption where needed
- Network firewalls where appropriate
- Security gateways provide firewalls and access control
- External data can be "pushed"
- Network and application intrusion detection
- Power System Attack Warning System

Toolkit Potential Applications

- Starter SCADA for small utilities
- Substation remote maintenance workstation
- SCADA with basic corporate interface
- Research testbed
- Application firewalls and access control gateways
- SCADA/control center with enhanced applications, interfaces, and security features
- Power System Attack Warning System

Overview of Possible Toolkit Security Controls

All toolkit platforms have a secure operating system, OS and application access control.


SCADA Network
FW
SCADA Server
FW
Corporate Network
FW
Substation Security Gateway
Control Center Network
FW
Corporate/ External Access Server
FW
Substation Internal Network
Security Workstation
Control Center Operations Workstations
External Data Security Gateway

Toolkit Development Status

- Concept and architecture developed
- Major open source software components selected
- Developmental components identified
- One critical component (MMS/OSI/RFC1006 stack) developed and initially tested
- Some other proof-of-concept prototypes and experiments
- Some HMI screen layouts defined

Use of SE-Linux in Toolkit

- Uses
  - Basic platform protection
  - Protect application access control
  - Confine data flows within servers and workstations
- Non-uses
  - Can't provide application access control based on network messages, application objects, or application business logic
  - Too complex for utility personnel to tailor and maintain to accommodate utility-defined roles and business rules
  - Policies not easily extended across networks

Challenges to SE-Linux development

- Better tools to facilitate roles defined by using entity
- Simplified support for mandatory access control within application objects and business process logic/rules on the same platform as the OS
- Methods of addressing mandatory access control within network applications (e.g., web services) where the user is known to the application but not to the OS
- Extension of common mandatory access control policies to multiple systems on a network