DESIGN, DEVELOPMENT AND ANALYSIS OF A COMPREHENSIVE OPEN SOURCE SYSTEM FOR PROACTIVE MANAGEMENT OF SECURITY ASPECTS OF A CONTROL NETWORK

S.S. Tomar[#], S.N. Chaudhari, H.S. Chouhan, V.K. Maurya, A. Rawat, RRCAT, Indore, India

Abstract

Control networks can only be assumed to be secure, when they work in complete isolation and all communication ports of the constituent control devices are disabled and are closely monitored for security breaches on 24X7 basis. With more and more control systems being developed using Common Out of The Shelf (COTS) computers, using windows OS, the chances of virus attacks on such control networks is extremely large. Handling zero day virus attacks or virus attacks with unknown cure, is a serious challenge for control network administrators. Another important aspect, somehow related to the security of the control network, is the rising temperatures of the control devices, because of 24X7 operation. All this is difficult to handle manually or using disconnected systems and hence there is a requirement of a comprehensive system which can do all this automatically. In this paper we will discuss the various security related parameters of the control networks and then present a simplified design followed by development details of a comprehensive open source system for proactive management of the security aspects of the control network.

INTRODUCTION

Modern day control networks are large and connected to Internet. They have distributed architecture and contain control, information & resource sharing related network components. COTS computers with Windows Operating System (OS) are used widely in large control networks with distributed architectures. Generally Windows OS based systems are prone to malware attacks. Due to this and the proximity of Internet to such networks, the security issues in such networks are numerous. Managing all these issues proactively and cost effectively is a huge challenge [1], but solutions in the form of numerous Free Open Source Software (FOSS) tools exist.

Out of numerous challenges in securing advanced distributed architecture [2], [3] control networks, proactive management of a) zero day malware attack and b) the overheating of control network components in distributed networks are important. The security concern about malware infected PCs on a network, is very serious, in control networks, since it sometimes causes transfer of control of the systems to some third party on Internet. The not so old “stuxnet” [4] and “duqu” menace are striking examples. The second security cum safety concern is the overheating of switching and server components, due to long continuous operations, in large distributed control networks. The situation caused by non working network communication channels and servers, due to overheating, in a control network can cause serious operational issues.

In this paper, we carry out risk analysis of a modern day Distributed Architecture Control Network (DACN) and then identify the areas for proactive management of its security aspects. We present a comprehensive approach to implement a secure DACN [5] using the various FOSS tools. This is followed by the design and development details of our in-house solutions, conceived, for addressing the above mentioned security concerns [6]. The detailed security analysis of the proposed comprehensive approach has been carried out and presented in the paper.

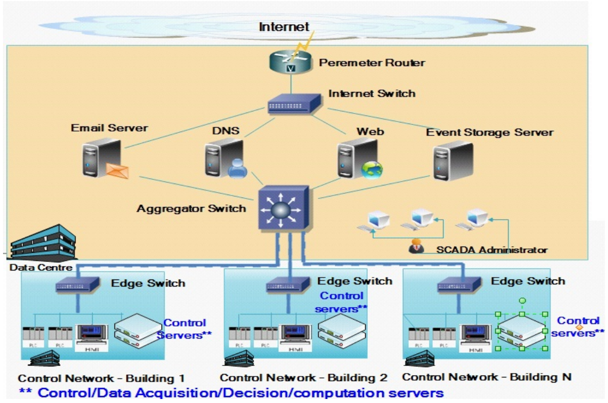

Figure 1: Distributed Architecture Control Network.

TYPICAL DACN

Figure 1 depicts the block diagram of a typical DACN which has the following salient features:

- Uses COTS PC, with windows OS, for computation and decision making.
- Uses Ethernet Technology with wide usage of Quality of Service (QoS) feature for real time communication with other systems.
- Uses modern communication servers like name, email and web for publishing information, thus is connected to Internet in some form or the other.
- Has some data acquisition and control hardware.
- Has some alarm generation system in place.
- Has a storage system for storing events.
- Has redundancy for Computation, Decision making, Communication and Data acquisition & Control.

[#] tomar@rrcat.gov.in

ISBN 978-3-95450-124-3

SECURITY ASPECTS OF DACN

The first step in managing the security aspects of any network [7] is the detailed risk analysis [8] of the setup. A detailed study of the DACN highlights the following risks factors:

- COTS PC with Windows OS, has risk of malwares changing the functionality of the system and in some cases transferring control of the system to third party.
- Ethernet technology based networks, spread over large campuses, have associated risks to Confidentiality, Integrity and Availability (CIA) of the data flowing through it.
- Mail/web/name servers have numerous risks in the form of malwares changing the functionality and data of these systems, SPAMs flooding email accounts and Domain Name System (DNS) poisoning, thereby causing Denial of Service (DoS) to legitimate users.
- Internet connection on a DACN poses all the risks towards CIA of information in the DACN.
- Distributed locations of the DACN components, have the associated risks of environmental changes getting unnoticed, thus leading to overheating and malfunctioning of systems.

Based on the risk analysis, we identify the areas for proactive management of the security aspects of a DACN connected to public network.

- Authentication Authorization Accounting (AAA)
- Firewall and DeMilitarized Zone (DMZ).
- Network Access Control (NAC).
- Network traffic encryption.
- Network traffic monitoring.
- Virus/SPAM management.
- Network fabric (Servers/Routers/Switches/PC /Software) management.
- Log monitoring and analysis.
- Alarm communication management.

SECURE DACN USING FOSS TOOLS

As with any network, the only design goal of a secure DACN is to make it secure by ensuring the CIA of information across various systems. Figure 2 depicts a very basic design of the secure DACN. The salient features of the design are as follows:

- A multi layered DMZ approach has been followed to restrict the access of servers over public networks, depending on the requirements.
- A proxy server based Internet access approach has been followed for internal users instead of the common direct Internet or Network Address Translation (NAT) approach to govern the web traffic flowing in/out of the DACN. This helps in providing access control and auditing every access to those networks.
- Filtering of web traffic for virus and SPAM has been done at every port of entry and exit of the traffic to/from public network.
- AAA of every access to the network has been incorporated in the design.
- Visibility of the resources, over public network has been reduced by using the perimeter level firewall in the design.
- Network traffic in the DACN has been encrypted at the network layer level.
- Network monitoring systems have been incorporated to monitor the traffic entering in/out of every DMZ, thus taking care of unusual/ unwanted traffic.
- Automated log analysis systems have been incorporated.
- Abnormal network event related real time multimode alarm generation & communication system has been incorporated.

To make the system cost effective, development of the system has been done using the below mentioned FOSS tools and Linux as OS.

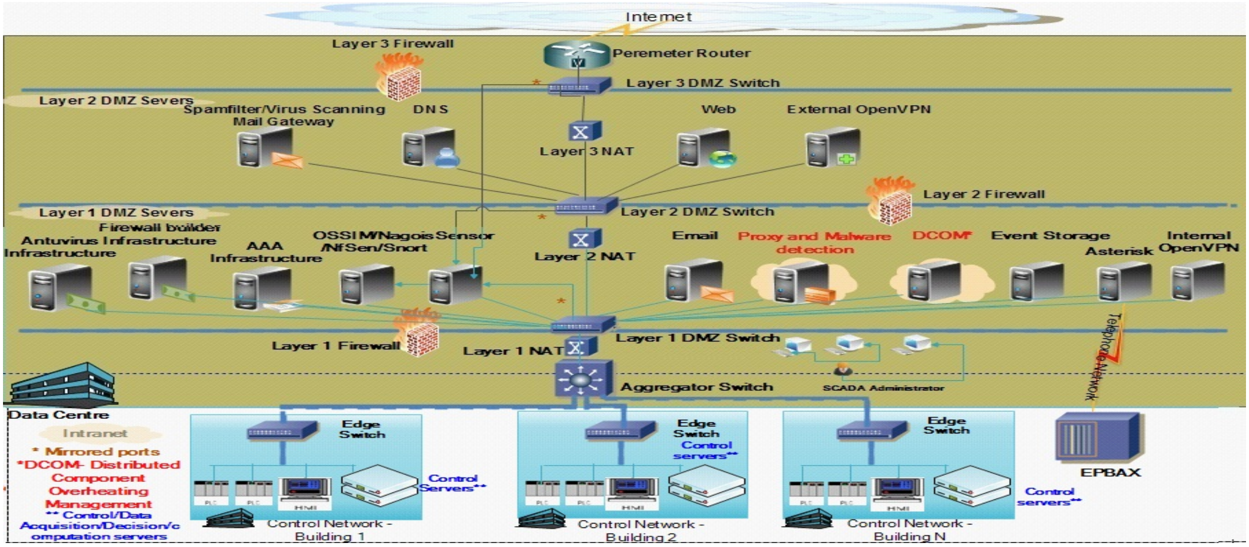

Figure 2: Securely designed modern day Distributed Architecture Control Network.

ISBN 978-3-95450-124-3

- AAA – CentOS Directory Server and FreeRadius.
- Firewall and DMZ – IPtables with Bastille and firewall builder is used to build and manage the IPtable rules.
- Network Access Control – PacketFence and Authenticated Squid Proxy server.
- Network traffic encryption – OpenVPN and Apache with secure hypertext transfer protocol.
- Network traffic monitoring – Traffic Capturing using mirroring and Netflow tools like ipcad on OpenBSD OS and Open Source Security Information Management (OSSIM) with nagios/snort, NfSen plugins.
- Virus/SPAM mgmt. – Clam AntiVirus (ClamAV) for virus filtering, SpamAssassin for SPAM control on servers. ClamAV and Microsoft Security Essentials on user end PCs.
- Network fabric management – OSSIM to monitor host and service availability. Bastille for server hardening and NetDisco for asset discovery.
- Log Analysis – Webalyzer for web logs and OSSEC with automated log analysis feature.
- Alarm communication system – Asterisk for communicating alarms using the telecommunication network, emails using email network, web server for status information display in the form of web pages.

In the following two subsections we discuss techniques to further enhance the security of this basic secure DACN by adding our in-house conceived, designed and developed systems.

a) Malware Detection System

In networks using squid proxy for access of Internet/Intranet, squid log analysis technique can be used to detect malware infected PCs on the network. The idea is that since the malwares like to proliferate & use Internet/network as the channel for proliferation hence on networks using proxy services, footprint of malwares are present in the proxy logs. We have designed & developed a complete system of automatically detecting suspected malware infected systems, blocking their access to Internet, informing the concerned users about the blocking and providing the user with a mechanism to unblock the PC for future Internet access after removal of the malware. Figure 3 illustrates the concept of the complete solution which consists of following subsystems.

- Malware infected PC detecting and blocking subsystem.
- User PC status checking subsystem.
- User controlled PC unblocking subsystem.

First of all the entire network is configured to use a squid proxy server for access to outside network/Internet.

The logging and the authentication options are enabled in the squid configuration file. Whenever a malware infected PC tries to access Internet, unauthenticated proxy requests are generated, which are marked with TCP_DENIED tags in the logs. The same tag is also generated for genuine unauthenticated requests of the users who may sometimes mistakenly give wrong passwords. But the difference in both the cases is that a genuine user may fail to give wrong password a limited/countable number of times only, while a malware infected PC tries and retries multiple times. This characteristic of the malware infected PC generating excessive TCP_DENIED log lines is exploited in this system to detect them. The routing table modification with a non routable entry for an IP is used to block the access of a particular PC/IP of the squid proxy server.

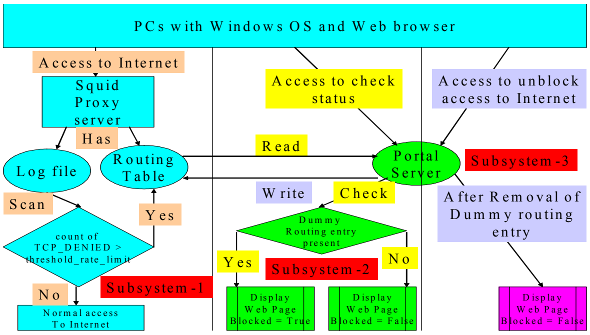
PCs with Windows OS and Web browser
Access to Internet
Squid Proxy server
Has
Log file
Routing Table
Scan
count of TCP_DENIED > threshold_rate_limit
Yes
No
Normal access To Internet
Subsystem-1
Access to check status
Read
Portal Server
Write
Check
Dummy Routing entry present
Yes
No
Subsystem-2
Display Web Page Blocked = True
Display Web Page Blocked = False
Access to unblock access to Internet
Subsystem-3
After Removal of Dummy routing entry
Display Web Page Blocked = False

Figure 3: Conceptual block diagram of the zero day Malware detection/blocking/unblocking solution.

Malware infected PC detecting and blocking subsystem consists of a script, in the squid proxy server, with the following algorithm:

Step 1. Define a threshold limit for generated TCP_DENIED requests for a PC/IP.
Step 2. Read the squid access log file for lines containing TCP_DENIED word. Only consider the lines that have got augmented since the last read.
Step 3. Read each line containing TCP_DENIED tag and count the number of such lines generated for each PC/IP.
Step 4. Compare the count of such lines with the threshold rate limit as defined in step 1.
Step 5. If the count of such lines generated by a PC/IP is more that the set threshold limit then mark the PC/IP as malware infected.
Step 6. For IPs listed in step 5, make a non routable entry in the routing table of the proxy server for each of them for blocking their access.

The design of the "user status checking subsystem" includes a portal with necessary web pages to publish the status of the user PC/IP. This subsystem is residing on another server, configured with a portal/web server, which is accessible to the PCs even when proxy access is not present. A script with the following algorithm is used to provide the required functionality:

Step 1. Get the IP details of the PC from where the status check function is being performed.
Step 2. Read the routing table of the proxy server which was modified by the detecting/blocking subsystem.
Step 3. Check for the routing table entry related to the PC/IP from where the status is being checked.
Step 4. If routing table entry for the IP exists then generate a web page showing Blocked = true as the PC status else generate a page with Blocked=false message

The design of the "User controlled PC unblocking subsystem" includes, extending the capabilities of the portal setup for the second subsystem with a script with the following algorithm:

Step 1. Get the IP of the PC from where the user requested for unblocking.
Step 2. Read the routing table of the proxy server.
Step 3. If the IP related routing entry is present in the proxy server's routing table then delete it.

b) Distributed Component Overheating Management System (DCOMS)

To prevent overheating of DACN components located at distant locations, this system has been designed and developed. It uses the Simple Network Management Protocol (SNMP) feature of the managed network switches. The idea is to continuously poll the switch for the value of the SNMP temperature variable and perform actions like generation of email alert and phone calls to the concerned administrators or shutting down of servers. Figure 4 illustrates the complete concept of the developed system.

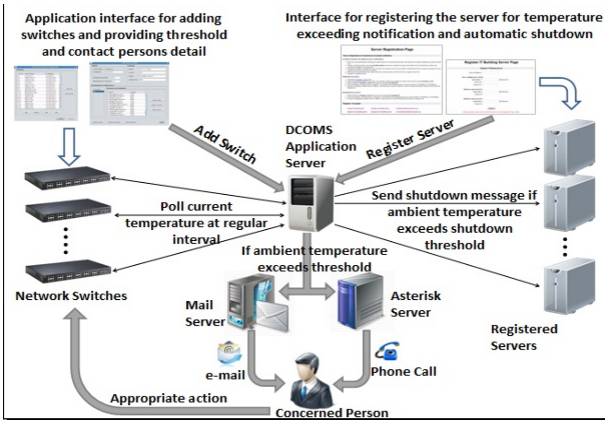
Application interface for adding switches and providing threshold and contact persons detail
Interface for registering the server for temperature exceeding notification and automatic shutdown
Add Switch
DCOMS Application Server
Register Server
Poll current temperature at regular interval
Send shutdown message if ambient temperature exceeds shutdown threshold
If ambient temperature exceeds threshold
Network Switches
Mail Server
Asterisk Server
Registered Servers
e-mail
Phone Call
Appropriate action
Concerned Person

Figure 4: Conceptual block diagram of the DCOMS.

The system design incorporates the following features and algorithms:

```
Step 1. Authorized persons can register network
        switches and servers to be monitored using the
        application/web interfaces. This information is
        stored in XML format.
Step 2. A program with the following algorithm executes
        in the DCOMS server:
    a. Read the database of registered switches and get
       IPs of the registered switches.
    b. For each switch IP.
        i. Connect to switch using the SNMP client and
           get SNMP variable value of the temperature
           variable.
        ii. Read the email-ids and phone numbers
            associated with the switches.
        iii. Read the shutdown threshold value of the
             switch
        iv. Read the registered server IPs associated with
            the switch
        v. If the current temperature exceeds switch
           overheating threshold then
            1. Send email to the concerned persons
               using the email server
            2. Generate a phone call to the concerned
               persons using the asterisk server
```

The design of the system for incorporating functionality of automatic server shutdown is:

```
Step 1. On the server (For registration, one time)
    a. Execute a program/script (register.sh) with the
       following algorithm:
        i. Create a restricted shell user in the server and
           copy the root public key of the DCOMS server
           to its authorized users list.
        ii. Install a script (shutdown_ server.sh) with the
            following algorithm:
            1. Read the message file as sent by DCOMS
               server.
            2. If the value of shutdown flag is true then
               initiate shutdown.
    b. Make an entry in cron scheduler to schedule the
       program for execution in every minute interval
    c. Make an entry in rc.local file to reset the shutdown
       flag value.
    d. Copy the DCOMS server root public key into the
       .ssh folder of the restricted shell user.
Step 2. On the DCOMS server (For every switch)
    a. If the current temperature of the associated switch
       exceeds server shutdown threshold then:
        i. Send shutdown message to the servers
        ii. Send email to the concerned persons using the
            email server
        iii. Generate a phone call to the concerned persons
             using the asterisk server.
Step 3. On the server (For deregistration, one time)
    a. Delete the cron entry related to
       shutdown_server.sh script
    b. Delete the rc.local entry for changing the
       shutdown flag.
    c. Delete the shutdown_server.sh script.
    d. Delete the restricted shell user.
```

For the development of malware detection system and DCOMS, Linux (CentOS 5.7) as OS, Apache as web server, JAVA for developing the switch threshold configurator, HTML and JavaScript for developing web interfaces, XML for storing information, CRON for scheduling, PHP, BASH scripts, *scp* for message passing to the remote hosts, SNMP for polling switches current temperature, Asterisk as telephony system and Qmail as mail server has been used.

SECURITY ANALYSIS

The analysis of the developed, secure DACN, is carried out with respect to the basic defining factors of security – CIA triad- of information in any network.

Confidentiality:

The AAA setup ensures that any flow of message on the network can be initiated only after proper authentication and authorization check. The Accounting feature of both the Directory Server and the RADIUS server provide the necessary audit trails of the accesses. For meeting the requirement of keeping the messages flowing in the DACN, between the various systems highly confidential, traffic encryption at network level is done by using the OpenVPN infrastructure. The use of authenticated squid proxy, enabled for logging, helps in auditing all the accesses made. The use of firewalls, DMZ, NAC and NAT feature provides access control and helps to meet the authorization objective.

Integrity:

Use of dual authentication scheme for authentication - as provided by OpenVPN and AAA infrastructure - by using digital certificates and passwords ensures integrity of the data flowing in the DACN. The AAA infrastructure provides all controls like password expiry, strong passwords, MD5 encrypted storage of passwords etc., to comply to ISO 27001 certification requirements. The use of antivirus software at every entry and exit point of the network ensures that the information published is unaltered because of malware attacks. The use of OSSEC, ensures continuous monitoring of the integrity of various important system configuration files. The use of Hash Message Authentication Code (HMAC) feature of OpenVPN also provides integrity checks of packets at network layer level.

Availability:

The malware detection system increases the availability of the DACN resources by reducing the chances of DoS. The DCOMS ensures continuous availability of DACN with healthy components. The use of network monitoring tools like OSSIM, NfSen, Nagios, Snort ensures that the resources are monitored continuously for failures and security compromising events. The use of OSSEC enhances availability by performing automated log analysis and generating alarms for unusual events. The use of Asterisk, Email and Web server provides functionality of communicating alarms using telecommunication network, email network and web respectively.

CONCLUSION

In a large distributed architecture network, foolproof security is impossible to achieve, thus its security aspects have to be managed. Modern day DACNs have more security requirements as compared to the resource & information sharing networks. In this paper we identified areas for proactive management of security aspects of the DACN. The basic secure DACN can be built using FOSS tools, which are freely available. The paper here presented a working set of FOSS tools available to develop a basic secure DACN.

In this paper we presented our approach to tackle two challenges existing in distributed architecture networks. The first one, of zero day malware attack, has been tackled by using the squid proxy server and log analysis automation technique. The entire process of log analysis automation, to generate alarms in case of malware detection has been presented. The second problem related to continuous monitoring of the distributed set of components for temperature changes have been handled using the available FOSS tools. The technique presented here provides a complete solution to proactively manage the distributed components for temperature variations at distributed locations.

REFERENCES

[1] Martin Naedele, “Addressing IT Security for Critical Control Systems,” Proceedings of the 40th Annual Hawaii International Conference on System Sciences, 07.

[2] Falk, R., “Industrial Sensor Network Security Architecture,” Fourth International Conference on Emerging Security Information Systems and Technologies (SECURWARE), 2010.

[3] “Architecture for SCADA Architecture for Secure SCADA and Distributed Control Systems Networks,” 2010, http://www.juniper.net/us/en/local/pdf/ whitepapers/2000276-en.pdf

[4] Karnouskos, S., “Stuxnet worm impact on industrial cyber-physical system security,” IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society, 7-10 Nov. 2011.

[5] Zhihu Wang, “Design and realization of computer network security perception control system,” IEEE 3rd International Conference on Communication Software and Networks (ICCSN), 27-29 May, 11, Nanning, China.

[6] Sopko, M. Winegardner, K. , “Process control network security concerns and remedies,” 2007 IEEE Cement Industry Technical Conference.

[7] Richard Kissel, Kevin Stine, Matthew Scholl, Hart Rossman, Jim Fahlsing , Jessica Gulick, NIST SP 800-64, “Security Considerations in the System Development Life Cycle,” Revision 2, 2008, http://csrc.nist.gov/publications/nistpubs/800-64-Rev2/SP800-64-Revision2.pdf

[8] Henryka Jormakka, Pekka Koponen, heimo Pentikainen, Henna Bartoszewicz Burczy, “Control systems of critical infrastructures, security analysis,” 2009, http://elektroenergetyka.pl/upload/file/2009/4/elektroenergetyka_nr_09_04_3.pdf