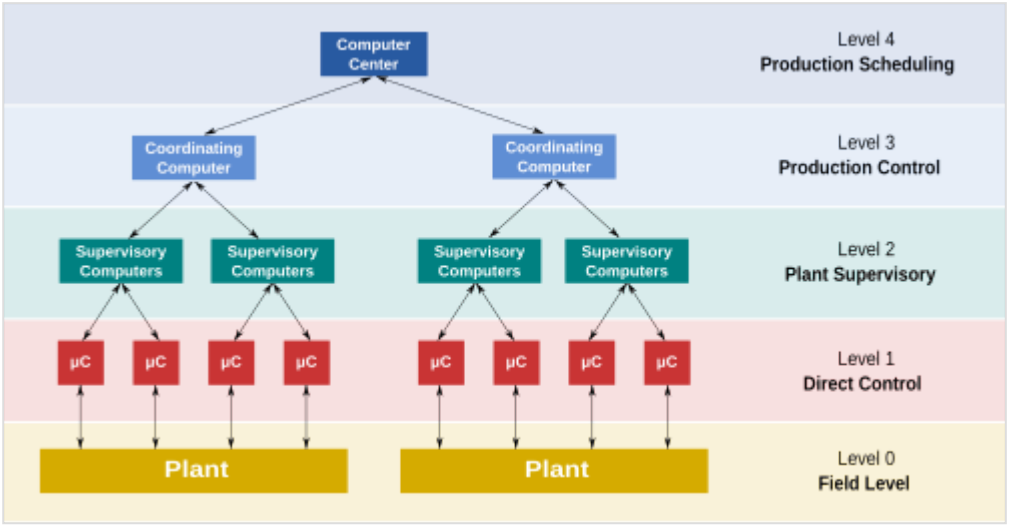
SCADA

SCADA (an acronym for **supervisory control and data acquisition**) is a control system architecture comprising computers, networked data communications and graphical user interfaces for high-level supervision of machines and processes. It also covers sensors and other devices, such as programmable logic controllers, which interface with process plant or machinery.

The operator interfaces which enable monitoring and the issuing of process commands, such as controller setpoint changes, are handled through the SCADA computer system. The subordinated operations, e.g. the real-time control logic or controller calculations, are performed by networked modules connected to the field sensors and actuators.

The SCADA concept was developed to be a universal means of remote-access to a variety of local control modules, which could be from different manufacturers and allowing access through standard automation protocols. In practice, large SCADA systems have grown to become similar to distributed control systems in function, while using multiple means of interfacing with the plant. They can control large-scale processes that can span multiple sites, and work over large distances. It is one of the most commonly-used types of industrial control systems.

Control operations



Functional levels of a manufacturing control operation

The key attribute of a SCADA system is its ability to perform a supervisory operation over a variety of other proprietary devices.

- Level 0 contains the field devices such as flow and temperature sensors, and final control elements, such as control valves.
- Level 1 contains the industrialized input/output (I/O) modules, and their associated distributed electronic processors.

- Level 2 contains the supervisory computers, which collate information from processor nodes on the system, and provide the operator control screens.
- Level 3 is the production control level, which does not directly control the process, but is concerned with monitoring production and targets.
- Level 4 is the production scheduling level.

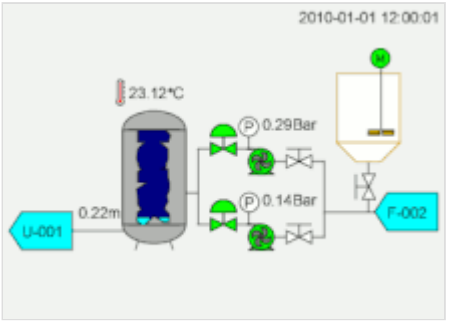
Level 1 contains the programmable logic controllers (PLCs) or remote terminal units (RTUs).

Level 2 contains the SCADA to readings and equipment status reports that are communicated to level 2 SCADA as required. Data is then compiled and formatted in such a way that a control room operator using the human-machine interface (HMI) can make supervisory decisions to adjust or override normal RTU (PLC) controls. Data may also be fed to a historian, often built on a commodity database management system, to allow trending and other analytical auditing.

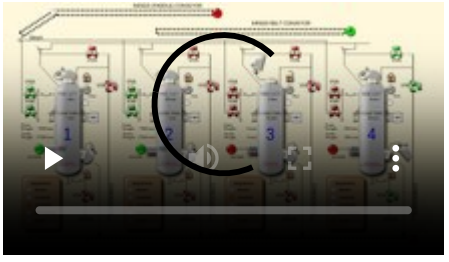
SCADA systems typically use a *tag database*, which contains data elements called *tags* or *points*, which relate to specific instrumentation or actuators within the process system. Data is accumulated against these unique process control equipment tag references.

Components

A SCADA system usually consists of the following main elements:



Typical SCADA mimic shown as an animation. For process plants, these are based upon the piping and instrumentation diagram.


More complex SCADA animation showing control of four batch cookers

Supervisory computers

This is the core of the SCADA system, gathering data on the process and sending control commands to the field connected devices. It refers to the computer and software responsible for communicating with the field connection controllers, which are RTUs and PLCs, and includes the HMI software running on operator workstations. In smaller SCADA systems, the supervisory computer may be composed of a single PC, in which case the HMI is a part of this computer. In larger SCADA systems, the master station may include several HMIs hosted on client computers, multiple servers for data acquisition, distributed software applications, and disaster recovery sites. To increase the integrity of the system the multiple servers will often be configured in a dual-redundant or hot-standby formation providing continuous control and monitoring in the event of a server malfunction or breakdown.

Remote terminal units

RTUs[1] connect to sensors and actuators in the process, and are networked to the supervisory computer system. RTUs have embedded control capabilities and often conform to the IEC 61131-3 standard for programming and support automation via ladder logic, a function block diagram or a variety of other languages. Remote locations often have little or no local infrastructure so it is not uncommon to find RTUs running off a small solar power system, using radio, GSM or satellite for communications, and being ruggedised to survive from -20C to +70C or even -40C to +85C without external heating or cooling equipment.

Programmable logic controllers

PLCs are connected to sensors and actuators in the process, and are networked to the supervisory system. In factory automation, PLCs typically have a high speed connection to the SCADA system. In remote applications, such as a large water treatment plant, PLCs may connect directly to SCADA over a wireless link, or more commonly, utilise an RTU for the communications management. PLCs are specifically designed for control and were the founding platform for the IEC 61131-3 programming languages. For economical reasons, PLCs are often used for remote sites where there is a large I/O count, rather than utilising an RTU alone.

Communication infrastructure

This connects the supervisory computer system to the RTUs and PLCs, and may use industry standard or manufacturer proprietary protocols. Both RTUs and PLCs operate autonomously on the near-real time control of the process, using the last command given from the supervisory system. Failure of the communications network does not necessarily stop the plant process controls, and on resumption of communications, the operator can continue with monitoring and control. Some critical systems will have dual redundant data highways, often cabled via diverse routes.

Human-machine interface

The HMI is the operator window of the supervisory system. It presents plant information to the operating personnel graphically in the form of mimic diagrams, which are a schematic representation of the plant being controlled, and alarm and event logging pages. The HMI is linked to the SCADA supervisory computer to provide live data to drive the mimic diagrams, alarm displays and trending graphs. In many installations the HMI is the graphical user interface for the operator, collects all data from external devices, creates reports, performs alarming, sends notifications, etc. Mimic diagrams consist of line graphics and schematic symbols to represent process elements, or may consist of digital photographs of the process equipment overlain with animated symbols. Supervisory operation of the plant is by means of the HMI, with operators issuing commands using mouse pointers, keyboards and touch screens. For example, a symbol of a pump can show the operator that the pump is running, and a flow meter symbol can show how much fluid it is pumping through the pipe. The operator can switch the pump off from the mimic by a mouse click or screen touch. The HMI will show the flow rate of the fluid in the pipe decrease in real time. The HMI package for a SCADA system typically includes a drawing program that the operators or system maintenance personnel use to change the way these points are represented in the interface. These representations can be as simple as an on-screen traffic light, which represents the state of an actual traffic light in the field, or as complex as a multi-projector display representing the position of all of the elevators in a skyscraper or all of the trains on a railway. A *historian* is a software service within the HMI which accumulates time-stamped data, events, and alarms in a database which can be queried or used to populate graphic trends in the HMI. The historian is a client that requests data from a data acquisition server.[2]

Alarm handling

An important part of most SCADA implementations is alarm handling. The system monitors whether certain alarm conditions are satisfied, to determine when an alarm event has occurred. Once an alarm event has been detected, one or more actions are taken (such as the activation of one or more alarm indicators, and perhaps the generation of email or text messages so that management or remote SCADA operators are informed). In many cases, a SCADA operator may have to acknowledge the alarm event; this may deactivate some alarm indicators, whereas other indicators remain active until the alarm conditions are cleared.

Alarm conditions can be explicit—for example, an alarm point is a digital status point that has either the value NORMAL or ALARM that is calculated by a formula based on the values in other analogue and digital points—or implicit: the SCADA system might automatically monitor whether the value in an analogue point lies outside high and low- limit values associated with that point.

Examples of alarm indicators include a siren, a pop-up box on a screen, or a coloured or flashing area on a screen (that might act in a similar way to the "fuel tank empty" light in a car); in each case, the role of the alarm indicator is to draw the operator's attention to the part of the system 'in alarm' so that appropriate action can be taken.

PLC/RTU programming

"Smart" RTUs, or standard PLCs, are capable of autonomously executing simple logic processes without involving the supervisory computer. They employ standardized control programming languages (such as those under IEC 61131-3, a suite of five programming languages including function block, ladder, structured text, sequence function charts and instruction list), that are frequently used to create programs which run on these RTUs and PLCs. Unlike a procedural language like C or FORTRAN, IEC 61131-3 has minimal training requirements by virtue of resembling historic physical control arrays. This allows SCADA system engineers to perform both the design and implementation of a program to be executed on an RTU or PLC.

A programmable automation controller (PAC) is a compact controller that combines the features and capabilities of a PC-based control system with that of a typical PLC. PACs are deployed in SCADA systems to provide RTU and PLC functions. In many electrical substation SCADA applications, "distributed RTUs" use information processors or station computers to communicate with digital protective relays, PACs, and other devices for I/O, and communicate with the SCADA master in lieu of a traditional RTU.

PLC commercial integration

Since about 1998, virtually all major PLC manufacturers have offered integrated HMI/SCADA systems, many of them using open and non-proprietary communications protocols. Numerous specialized third-party HMI/SCADA packages, offering built-in compatibility with most major PLCs, have also entered the market, allowing mechanical engineers, electrical engineers and technicians to configure HMIs themselves, without the need for a custom-made program written by a software programmer. The Remote Terminal Unit (RTU) connects to physical equipment. Typically, an RTU converts the electrical signals from the equipment to digital values. By converting and sending these electrical signals out to equipment the RTU can control equipment.

Communication infrastructure and methods

SCADA systems have traditionally used combinations of radio and direct wired connections, although SONET/SDH is also frequently used for large systems such as railways and power stations. The remote management or monitoring function of a SCADA system is often referred to as telemetry. Some users

want SCADA data to travel over their pre-established corporate networks or to share the network with other applications. The legacy of the early low-bandwidth protocols remains, though.

SCADA protocols are designed to be very compact. Many are designed to send information only when the master station polls the RTU. Typical legacy SCADA protocols include Modbus RTU, RP-570, Profibus and Conitel. These communication protocols, with the exception of Modbus (Modbus has been made open by Schneider Electric), are all SCADA-vendor specific but are widely adopted and used. Standard protocols are IEC 60870-5-101 or 104, IEC 61850 and DNP3. These communication protocols are standardized and recognized by all major SCADA vendors. Many of these protocols now contain extensions to operate over TCP/IP. Although the use of conventional networking specifications, such as TCP/IP, blurs the line between traditional and industrial networking, they each fulfill fundamentally differing requirements.[3] Network simulation can be used in conjunction with SCADA simulators to perform various 'what-if' analyses.

With increasing security demands (such as North American Electric Reliability Corporation (NERC) and critical infrastructure protection (CIP) in the US), there is increasing use of satellite-based communication. This has the key advantages that the infrastructure can be self-contained (not using circuits from the public telephone system), can have built-in encryption, and can be engineered to the availability and reliability required by the SCADA system operator. Earlier experiences using consumer-grade VSAT were poor. Modern carrier-class systems provide the quality of service required for SCADA.[4]

RTUs and other automatic controller devices were developed before the advent of industry wide standards for interoperability. The result is that developers and their management created a multitude of control protocols. Among the larger vendors, there was also the incentive to create their own protocol to "lock in" their customer base. A list of automation protocols is compiled here.

An example of efforts by vendor groups to standardize automation protocols is the OPC-UA (formerly "OLE for process control" now Open Platform Communications Unified Architecture).

Architecture development

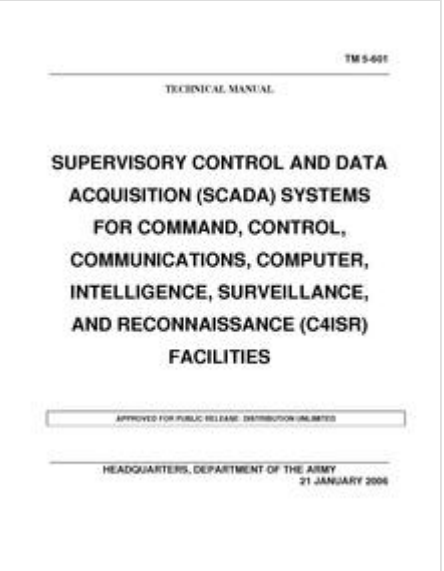
SCADA systems have evolved through four generations as follows:[5][6][7][8]

Early SCADA system computing was done by large minicomputers. Common network services did not exist at the time SCADA was developed. Thus SCADA systems were independent systems with no connectivity to other systems. The communication protocols used were strictly proprietary at that time. The first-generation SCADA system redundancy was achieved using a back-up mainframe system connected to all the Remote Terminal Unit sites and was used in the event of failure of the primary mainframe system.[9] Some first generation SCADA systems were developed as "turn key" operations that ran on minicomputers such as the PDP-11 series.[10]

SCADA information and command processing were distributed across multiple stations which were connected through a LAN. Information was shared in near real time. Each station was responsible for a particular task, which reduced the cost as compared to First Generation SCADA. The network protocols

used were still not standardized. Since these protocols were proprietary, very few people beyond the developers knew enough to determine how secure a SCADA installation was. Security of the SCADA installation was usually overlooked.

Similar to a distributed architecture, any complex SCADA can be reduced to the simplest components and connected through communication protocols. In the case of a networked design, the system may be spread across more than one LAN network called a process control network (PCN) and separated geographically. Several distributed architecture SCADAs running in parallel, with a single supervisor and historian, could be considered a network architecture. This allows for a more cost-effective solution in very large scale systems.


TM 5-601

TECHNICAL MANUAL

SUPERVISORY CONTROL AND DATA ACQUISITION (SCADA) SYSTEMS FOR COMMAND, CONTROL, COMMUNICATIONS, COMPUTER, INTELLIGENCE, SURVEILLANCE, AND RECONNAISSANCE (C4ISR) FACILITIES

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

HEADQUARTERS, DEPARTMENT OF THE ARMY
21 JANUARY 2006

The United States Army's Training Manual 5-601 covers "SCADA Systems for C4ISR Facilities"

The growth of the internet has led SCADA systems to implement web technologies allowing users to view data, exchange information and control processes from anywhere in the world through web SOCKET connection.[11][12] The early 2000s saw the proliferation of Web SCADA systems.[13][14][15] Web SCADA systems use web browsers such as Google Chrome and Mozilla Firefox as the graphical user interface (GUI) for the operators HMI.[16][13] This simplifies the client side installation and enables users to access the system from various platforms with web browsers such as servers, personal computers, laptops, tablets and mobile phones.[13][17]

Security

SCADA systems that tie together decentralized facilities such as power, oil, gas pipelines, water distribution and wastewater collection systems were designed to be open, robust, and easily operated and repaired, but not necessarily secure.[18][19] The move from proprietary technologies to more standardized and open solutions together with the increased number of connections between SCADA systems, office networks and the Internet has made them more vulnerable to types of network attacks that are relatively common in computer security. For example, United States Computer Emergency Readiness Team (US-CERT) released a vulnerability advisory[20] warning that unauthenticated users could download sensitive configuration information including password hashes from an Inductive Automation Ignition system utilizing a standard attack type leveraging access to the Tomcat Embedded Web server. Security researcher Jerry Brown submitted a similar advisory regarding a buffer overflow vulnerability[21] in a Wonderware InBatchClient ActiveX control. Both vendors made updates available prior to public vulnerability release. Mitigation recommendations were standard patching practices and requiring VPN access for secure connectivity. Consequently, the security of some SCADA-based systems has come into question as they are seen as potentially vulnerable to cyber attacks.[22][23][24]

In particular, security researchers are concerned about:

- The lack of concern about security and authentication in the design, deployment and operation of some existing SCADA networks
- The belief that SCADA systems have the benefit of security through obscurity through the use of specialized protocols and proprietary interfaces

- The belief that SCADA networks are secure because they are physically secured
- The belief that SCADA networks are secure because they are disconnected from the Internet

SCADA systems are used to control and monitor physical processes, examples of which are transmission of electricity, transportation of gas and oil in pipelines, water distribution, traffic lights, and other systems used as the basis of modern society. The security of these SCADA systems is important because compromise or destruction of these systems would impact multiple areas of society far removed from the original compromise. For example, a blackout caused by a compromised electrical SCADA system would cause financial losses to all the customers that received electricity from that source. How security will affect legacy SCADA and new deployments remains to be seen.

There are many threat vectors to a modern SCADA system. One is the threat of unauthorized access to the control software, whether it is human access or changes induced intentionally or accidentally by virus infections and other software threats residing on the control host machine. Another is the threat of packet access to the network segments hosting SCADA devices. In many cases, the control protocol lacks any form of cryptographic security, allowing an attacker to control a SCADA device by sending commands over a network. In many cases SCADA users have assumed that having a VPN offered sufficient protection, unaware that security can be trivially bypassed with physical access to SCADA-related network jacks and switches. Industrial control vendors suggest approaching SCADA security like Information Security with a defense in depth strategy that leverages common IT practices.[25] Apart from that, research has shown that the architecture of SCADA systems has several other vulnerabilities, including direct tampering with RTUs, communication links from RTUs to the control center, and IT software and databases in the control center.[26] The RTUs could, for instance, be targets of deception attacks injecting false data [27] or denial-of-service attacks.

The reliable function of SCADA systems in our modern infrastructure may be crucial to public health and safety. As such, attacks on these systems may directly or indirectly threaten public health and safety. Such an attack has already occurred, carried out on Maroochy Shire Council's sewage control system in Queensland, Australia.[28] Shortly after a contractor installed a SCADA system in January 2000, system components began to function erratically. Pumps did not run when needed and alarms were not reported. More critically, sewage flooded a nearby park and contaminated an open surface-water drainage ditch and flowed 500 meters to a tidal canal. The SCADA system was directing sewage valves to open when the design protocol should have kept them closed. Initially this was believed to be a system bug. Monitoring of the system logs revealed the malfunctions were the result of cyber attacks. Investigators reported 46 separate instances of malicious outside interference before the culprit was identified. The attacks were made by a disgruntled ex-employee of the company that had installed the SCADA system. The ex-employee was hoping to be hired by the utility full-time to maintain the system.

In April 2008, the Commission to Assess the Threat to the United States from Electromagnetic Pulse (EMP) Attack issued a Critical Infrastructures Report which discussed the extreme vulnerability of SCADA systems to an electromagnetic pulse (EMP) event. After testing and analysis, the Commission concluded: "SCADA systems are vulnerable to EMP insult. The large numbers and widespread reliance on such systems by all of the Nation’s critical infrastructures represent a systemic threat to their continued operation following an EMP event. Additionally, the necessity to reboot, repair, or replace large numbers of geographically widely dispersed systems will considerably impede the Nation’s recovery from such an assault."[29]

Many vendors of SCADA and control products have begun to address the risks posed by unauthorized access by developing lines of specialized industrial firewall and VPN solutions for TCP/IP-based SCADA networks as well as external SCADA monitoring and recording equipment. The International Society of Automation (ISA) started formalizing SCADA security requirements in 2007 with a working group, WG4. WG4 "deals specifically with unique technical requirements, measurements, and other features required to evaluate and assure security resilience and performance of industrial automation and control systems devices".[30]

The increased interest in SCADA vulnerabilities has resulted in vulnerability researchers discovering vulnerabilities in commercial SCADA software and more general offensive SCADA techniques presented to the general security community.[31] In electric and gas utility SCADA systems, the vulnerability of the large installed base of wired and wireless serial communications links is addressed in some cases by applying bump-in-the-wire devices that employ authentication and Advanced Encryption Standard encryption rather than replacing all existing nodes.[32]

In June 2010, anti-virus security company VirusBlokAda reported the first detection of malware that attacks SCADA systems (Siemens' WinCC/PCS 7 systems) running on Windows operating systems. The malware is called Stuxnet and uses four zero-day attacks to install a rootkit which in turn logs into the SCADA's database and steals design and control files.[33][34] The malware is also capable of changing the control system and hiding those changes. The malware was found on 14 systems, the majority of which were located in Iran.[35]

In October 2013 *National Geographic* released a docudrama titled *American Blackout* which dealt with an imagined large-scale cyber attack on SCADA and the United States' electrical grid.[36]

Uses

Both large and small systems can be built using the SCADA concept. These systems can range from just tens to thousands of control loops, depending on the application. Example processes include industrial, infrastructure, and facility-based processes, as described below:

Example of SCADA used in office environment to remotely monitor a process

- Industrial processes include manufacturing, process control, power generation, fabrication, and refining, and may run in continuous, batch, repetitive, or discrete modes.
- Infrastructure processes may be public or private, and include water treatment and distribution, wastewater collection and treatment, oil and gas pipelines, electric power transmission and distribution, and wind farms.
- Facility processes, including buildings, airports, ships, and space stations. They monitor and control heating, ventilation, and air conditioning systems (HVAC), access, and energy consumption.

However, SCADA systems may have security vulnerabilities, so the systems should be evaluated to identify risks and solutions implemented to mitigate those risks.[37]

See also

- DNP3 – Computer network protocol
- IEC 60870
- EPICS – Software infrastructure for building distributed control systems

References

1. Jeff Hieb (2008). *Security Hardened Remote Terminal Units for SCADA Networks* (https://books.google.com/books?id=A-_PPgAACAAJ). University of Louisville.
2. Aquino-Santos, Raul (30 November 2010). *Emerging Technologies in Wireless Ad-hoc Networks: Applications and Future Development: Applications and Future Development* (https://books.google.com/books?id=UN2eBQAAQBAJ&pg=PA43). IGI Global. pp. 43–. ISBN 978-1-60960-029-7.
3. "Introduction to Industrial Control Networks" (http://www.rfidblog.org.uk/Preprint-GallowayHancke-IndustrialControlSurvey.pdf) (PDF). *IEEE Communications Surveys and Tutorials*. 2012.
4. Bergan, Christian (August 2011). "Demystifying Satellite for the Smart Grid: Four Common Misconceptions" (https://web.archive.org/web/20120331210553/http://www.elp.com/index/display/article-display/5666163079/articles/utility-automation-engineering-td/volume-16/issue-8/features/demystifying-satellite-for-the-smart-grid-four-common-misconceptions.html). *Electric Light & Power*. Utility Automation & Engineering T&D. **16** (8). Tulsa, OK: PennWell. Four. Archived from the original (http://www.elp.com/index/display/article-display/5666163079/articles/utility-automation-engineering-td/volume-16/issue-8/features/demystifying-satellite-for-the-smart-grid-four-common-misconceptions.html) on 31 March 2012. Retrieved 2 May 2012. "satellite is a cost-effective and secure solution that can provide backup communications and easily support core smart grid applications like SCADA, telemetry, AMI backhaul and distribution automation"
5. OFFICE OF THE MANAGER NATIONAL COMMUNICATIONS SYSTEM (October 2004). "Supervisory Control and Data Acquisition (SCADA) Systems" (https://web.archive.org/web/20150714225002/https://scadahacker.com/library/Documents/ICS_Basics/SCADA%20Basics%20-%20NCS%20TIB%2004-1.pdf) (PDF). NATIONAL COMMUNICATIONS SYSTEM. Archived from the original (https://scadahacker.com/library/Documents/ICS_Basics/SCADA%20Basics%20-%20NCS%20TIB%2004-1.pdf) (PDF) on 14 July 2015. Retrieved 14 July 2015.
6. "SCADA Systems april 2014" (http://www.engineersgarage.com/articles/scada-systems).
7. J. Russel. "A Brief History of SCADA/EMS (2015)" (https://web.archive.org/web/20150811051350/http://scadahistory.com/). Archived from the original (http://scadahistory.com/) on 11 August 2015.
8. Abbas, H.A. (2014). Future SCADA challenges and the promising solution: the agent-based SCADA. IJCIS, 10, 307-333.
9. *Security Hardened Remote Terminal Units for SCADA Networks* (https://books.google.com/books?id=d5O5N1vjbQgC&pg=PA12). 2008. pp. 12–. ISBN 978-0-549-54831-7.
10. UJVAROSI, Alexandru (2 November 2016). "EVOLUTION OF SCADA SYSTEMS" (https://web.archive.org/web/20211028142517/https://webbut.unitbv.ro/BU2015/Series%20I/2016/BULETIN%20I%20PDF/Ujvarosi_Al.pdf) (PDF). Archived from the original (http://webbut.unitbv.ro/BU2015/Series%20I/2016/BULETIN%20I%20PDF/Ujvarosi_Al.pdf) (PDF) on 28 October 2021.

11. *R. Fan, L. Cheded and O. Toker, "Internet-based SCADA: a new approach using Java and XML," in Computing & Control Engineering Journal, vol. 16, no. 5, pp. 22-26, Oct.-Nov. 2005*, S2CID 62150803 (https://api.semanticscholar.org/CorpusID:62150803)
12. *R. J. Robles and T. H. Kim, "Architecture for SCADA with Mobile Remote Components", Proceedings of the 12th WSEAS International Conference on Automatic Control, Modelling & Simulation* (https://dl.acm.org/doi/10.5555/1844174.1844240), 29 May 2010, pp. 346–350, ISBN 978-954-92600-1-4 – via dl.acm.org
13. *Abbas, H.A. and Mohamed, A.M. (2011) 'Review on the design of web based SCADA systems based on OPC DA protocol', International Journal of Computer Networks, February, Vol. 2, No. 6, pp.266–277, Malaysia*, S2CID 18743659 (https://api.semanticscholar.org/CorpusID:18743659)
14. *Qiu B, Gooi HB. Web-based scada display systems (wsds) for access via internet. Power Systems, IEEE Transactions on 2000;15(2):681–686* (https://ieeexplore.ieee.org/document/867159), doi:10.1109/59.867159 (https://doi.org/10.1109%2F59.867159) – via ieeexplore.ieee.org
15. *Li D, Serizawa Y, Kiuchi M. Concept design for a web-based supervisory control and data-acquisition (scada) system. In: Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES; Vol. 1; p. 32–36*, doi:10.1109/TDC.2002.1178256 (https://doi.org/10.1109%2FTDC.2002.1178256), S2CID 113523881 (https://api.semanticscholar.org/CorpusID:113523881)
16. *Kovaliuk, D. O., Huza, K. M., & Kovaliuk, O. O. (2018). Development of SCADA System based on Web Technologies. International Journal of Information Engineering and Electronic Business (IJIEEB), 10(2), 25-32*, doi:10.5815/IJIEEB.2018.02.04 (https://doi.org/10.5815%2FIJIEEB.2018.02.04), S2CID 65360293 (https://api.semanticscholar.org/CorpusID:65360293)
17. *J. M. Lynch, "An Internet Based SCADA System", BSc Project Report, University of Southern Queensland, Queensland, Oct. 2005*, S2CID 109628360 (https://api.semanticscholar.org/CorpusID:109628360)
18. Boyes, Walt (2011). *Instrumentation Reference Book, 4th Edition*. USA: Butterworth-Heinemann. p. 27. ISBN 978-0-7506-8308-1.
19. Siggins, Morgana. "14 Major SCADA Attacks and What You Can Learn From Them" (https://www.dpstele.com/blog/major-scada-hacks.php). *DPS Telecom*. Retrieved 26 April 2021.
20. "ICSA-11-231-01—INDUCTIVE AUTOMATION IGNITION INFORMATION DISCLOSURE VULNERABILITY" (https://web.archive.org/web/20121105230108/http://www.us-cert.gov/control_systems/pdf/ICSA-11-231-01.pdf) (PDF). 19 August 2011. Archived from the original (http://www.us-cert.gov/control_systems/pdf/ICSA-11-231-01.pdf) (PDF) on 5 November 2012. Retrieved 21 January 2013.
21. "ICSA-11-094-01—WONDERWARE INBATCH CLIENT ACTIVEX BUFFER OVERFLOW" (https://ics-cert.us-cert.gov/pdf/ICSA-11-094-01.pdf) (PDF). 13 April 2011. Retrieved 26 March 2013.
22. "Cyberthreats, Vulnerabilities and Attacks on SCADA Networks" (https://web.archive.org/web/20120813015252/http://gspp.berkeley.edu/iths/Tsang_SCADA%20Attacks.pdf) (PDF). *Rosa Tang, berkeley.edu*. Archived from the original (http://gspp.berkeley.edu/iths/Tsang_SCADA%20Attacks.pdf) (PDF) on 13 August 2012. Retrieved 1 August 2012.
23. D. Maynor and R. Graham (2006). "SCADA Security and Terrorism: We're Not Crying Wolf" (https://www.blackhat.com/presentations/bh-federal-06/BH-Fed-06-Maynor-Graham-up.pdf) (PDF).
24. Robert Lemos (26 July 2006). "SCADA system makers pushed toward security" (http://www.securityfocus.com/news/11402). SecurityFocus. Retrieved 9 May 2007.
25. "Industrial Security Best Practices" (http://www.rockwellautomation.com/resources/downloads/rockwellautomation/pdf/products-technologies/security-technology/securat001aene.pdf) (PDF). Rockwell Automation. Retrieved 26 March 2013.

26. Giani, A.; Sastry, S.; Johansson, H.; Sandberg, H. (2009). "The VIKING project: An initiative on resilient control of power networks". *2009 2nd International Symposium on Resilient Control Systems* (http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-80021). IEEE. pp. 31–35. doi:10.1109/ISRCS.2009.5251361 (https://doi.org/10.1109%2FISRCS.2009.5251361). ISBN 978-1-4244-4853-1. S2CID 14917254 (https://api.semanticscholar.org/CorpusID:14917254).
27. Liu, Y.; Ning, P.; Reiter, MK. (May 2011). "False Data Injection Attacks against State Estimation in Electric Power Grids". *ACM Transactions on Information and System Security*. Vol. 14. Association for Computing Machinery. pp. 1–33. doi:10.1145/1952982.1952995 (https://doi.org/10.1145%2F1952982.1952995). S2CID 2305736 (https://api.semanticscholar.org/CorpusID:2305736).
28. Slay, J.; Miller, M. (November 2007). "Chpt 6: Lessons Learned from the Maroochy Water Breach". *Critical infrastructure protection* (Online-Ausg. ed.). Springer Boston. pp. 73–82. doi:10.1007/978-0-387-75462-8_6 (https://doi.org/10.1007%2F978-0-387-75462-8_6). ISBN 978-0-387-75461-1.
29. http://www.empcommission.org/docs/A2473-EMP_Commission-7MB.pdf Report of the Commission to Assess the Threat to the United States from Electromagnetic Pulse (EMP) Attack (http://www.empcommission.org/docs/A2473-EMP_Commission-7MB.pdf) (PDF) (Report). April 2008. p. 9. Retrieved 31 May 2024.
30. "Security for all" (http://www.isa.org). *InTech*. June 2008. Retrieved 2 May 2012.
31. "SCADA Security – Generic Electric Grid Malware Design" (https://web.archive.org/web/20090107085040/http://www.c4-security.com/SCADA%20Security%20-%20Generic%20Electric%20Grid%20Malware%20Design%20-%20SyScan08.pps). Archived from the original (http://www.c4-security.com/SCADA%20Security%20-%20Generic%20Electric%20Grid%20Malware%20Design%20-%20SyScan08.pps) on 7 January 2009.
32. KEMA, Inc (November 2006). "Substation Communications: Enabler of Automation" (https://web.archive.org/web/20071103173939/http://www.utc.org/?p=33398). Utilities Telecom Council. pp. 3–21. Archived from the original (http://www.utc.org/?p=33398) on 3 November 2007. Retrieved 19 January 2022.
33. Mills, Elinor (21 July 2010). "Details of the first-ever control system malware (FAQ)" (http://news.cnet.com/8301-27080_3-20011159-245.html). *CNET*. Retrieved 21 July 2010.
34. "SIMATIC WinCC / SIMATIC PCS 7: Information concerning Malware / Virus / Trojan" (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=43876783&caller=view). Siemens. 21 July 2010. Retrieved 22 July 2010. "malware (trojan) which affects the visualization system WinCC SCADA."
35. "Siemens: Stuxnet worm hit industrial systems" (https://archive.today/20120525053210/http://www.computerworld.com/s/article/print/9185419/Siemens_Stuxnet_worm_hit_industrial_systems?taxonomyName=Network+Security&taxonomyId=142). Archived from the original (http://www.computerworld.com/s/article/print/9185419/Siemens_Stuxnet_worm_hit_industrial_systems?taxonomyName=Network+Security&taxonomyId=142) on 25 May 2012. Retrieved 16 September 2010.
36. "American Blackout" (https://web.archive.org/web/20150313204218/http://channel.nationalgeographic.com/american-blackout/). National Geographic Channel. Archived from the original (http://channel.nationalgeographic.com/american-blackout/) on 13 March 2015. Retrieved 14 October 2016.
37. Boyer, Stuart A. (2010). *SCADA Supervisory Control and Data Acquisition*. USA: ISA - International Society of Automation. p. 179. ISBN 978-1-936007-09-7.

External links

- UK SCADA security guidelines (https://web.archive.org/web/20130620125607/http://www.cpni.gov.uk/advice/cyber/scada/)
- BBC NEWS | Technology | Spies 'infiltrate US power grid' (http://news.bbc.co.uk/1/hi/technology/7990997.stm)