Secure HTTPS Provides Enhanced Security in a Building Management System

CONTEMPORARY CONTROLS

Network Security is more critical than ever in today's building management system (BMS) networks to ensure authentication, integrity, and confidentiality of data transferred over the Internet. This white paper describes how BACnet-complaint devices that incorporate HTTPS deliver encrypted communication and protect the integrity of client data. This white paper also describes the HTTPS authentication and encryption method which utilizes keys and digital certificates. It compares certificates generated by a Certificate Authority (CA) vs. self-signed certificates and provides a resource to create your own self-signed certificate.

BACnet remains the most popular protocol utilized in HVACR control systems and there is a robust ecosystem of devices that comprise these systems, including Gateways to integrate other protocols, such as Modbus and EnOcean, to BACnet. As more and more devices are utilized to meet the demands of today's building management system (BMS) and smart building infrastructures, network security is more critical than ever to ensure authentication, integrity, and confidentiality of data transferred over the Internet.

BACnet-complaint devices that incorporate HTTPS (Secure HTTP) deliver encrypted communication and protect the integrity of client data. Resident HTTPS webservers allow commissioning, status reporting, and troubleshooting in a secure manner using any standard web browser, thereby improving access control to the devices.

HTTPS (Secure HTTP) uses encryption for secure communication over an IP network. HTTPS traffic is encrypted using Transport Layer Security (TLS), formerly Secure Sockets Layer (SSL). The protocol is still referred to as HTTP over SSL, commonly shown as **https://** in the browser address bar.

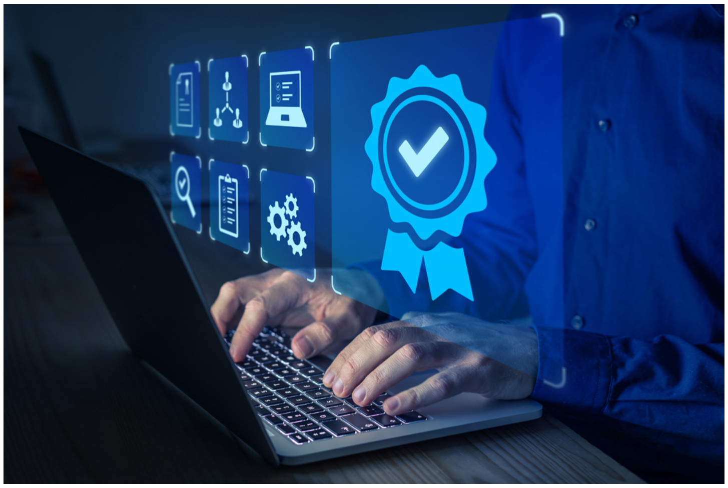
Digital Certificates

SSL/TLS relies on the use of keys and digital certificates for data encryption, device authentication, and data integrity. Keys occur in pairs (public/private) and are used for encryption/decryption. A public key is used for encryption, while the private key is used for decryption.

Digital certificates are used for authentication and encryption, verifying ownership and authenticity to ensure that only authorized devices communicate with each other. The public key is part of the certificate, while the private key is secret to the device. Mechanisms exist to generate certificates and keys for a device and to scale the architecture to multiple devices.

Digital Certificates—Certificate Authority

Certificates are typically issued and managed by a trusted third-party company, called a Certificate Authority (CA). Getting an SSL certificate installed for a website by a well-known CA that is trusted by all devices and browsers, such as DigiCert, Comodo, GoDaddy, Lets Encrypt, can provide access to the website seamlessly over the public Internet. The device can get the certificate directly from the CA or send a Certificate Signing Request (CSR) to the CA to get the corresponding certificate. These trusted CAs only provide certificates to websites or devices which have a public IP address. They won't provide certificates for devices on an internal network with private IP addresses.



Digital Certificates— Public Key Infrastructure

For an internal BMS network, getting a certificate from a public CA is not necessary and can be expensive given the considerable number of devices in a building. The IT department can implement their own infrastructure to generate these keys and certificates. The term PKI (Public Key Infrastructure) is used to define this setup. The building automation product vendors may also have specific software tools to implement the PKI, but the certificates and keys for all devices at a site, irrespective of their brand, must be generated from the same tool to ensure interoperability. The certificates on devices also expire and need to be renewed.

Devices used on internal networks can also employ a self-signed digital certificate to make a web browser trust your internal devices. A self-signed certificate is a type of SSL/TLS credential you sign yourself rather than having it signed by a trusted third-party CA. If you don't have an IT department, you can generate the self-signed certificate yourself. In addition, generating a self-signed certificate for internal network devices eliminates the associated cost of getting a certificate from a trusted third-party CA.

Digital Certificates— Self-signed

Self-signed digital certificates are created by signing the certificate with the owner's private key. They are created, issued, and signed by the company or developer who is responsible for the website/software being signed. Unlike certificates issued by a trusted CA, no external party verifies a self-signed certificate. Self-signed certificates are fast, free, and easy to issue. They are appropriate for local development, testing, or staging environments, internal network websites and providing secure webpages for devices. However, you must be aware of their limitations, such as despite the strong encryption they provide, they lack the backing of recognized authority, so browsers on different PCs will display security warnings for them.

Digital Certificates – OpenSSL

You can generate and install a self-signed certificate using OpenSSL, a commonly used command-line utility for generating keys, creating certificate signing requests (CSRs), and managing certificates.

According to OpenSSL documentation at https://docs.openssl.org/master/man7/ossl-guide-introduction/ *"OpenSSL is a robust, commercial-grade, full-featured toolkit for general-purpose cryptography and secure communication. Its features are made available via a command line application that enables users to perform various cryptography related functions such as generating keys and certificates. Additionally, it supplies two libraries that application developers can use to implement cryptography-based capabilities and to securely communicate across a network. Finally, it also has a set of providers that supply implementations of a broad set of cryptographic algorithms.*

OpenSSL is fully open source. Version 3.0 and above are distributed under the Apache v2 license."

If you don't have OpenSSL on your Windows's PC, you can install an OpenSSL package. If you are accessing the HTTPS device from a different PC, a security warning message will appear. You must download the self-signed certificate and install it to your local machine's trusted certificate store.

For more information, Contemporary Controls has created an Application Note: How to Create and Use Self-Signed SSL Certificates that explains how to add OpenSSL and create a self-signed certificate for Windows using Windows Package Manager, WinGet. WinGet is a free and open-source package manager designed by Microsoft that allows users to discover, install, upgrade, remove, and configure applications on Windows 10, Windows 11, and Windows Server 2025 computers. The application note also explains how to install this self-signed certificate on the device, and how to download and install the self-signed certificate on different Windows machines to eliminate the security warning. Instructions are provided for commonly used browsers—Google Chrome, Microsoft Edge, and Mozilla Firefox—and how to overcome the Security Warning message.

Conclusion

HTTPS encrypts the transport of data to ensure data integrity and prevents information from being modified, corrupted, or stolen during transmission. SSL/TLS protocols authenticate users to secure information and ensure it won't be revealed to unauthorized users. HTTPS requires digital certificates to validate the domain ownership and integrity. For external networks, you should obtain this credential from a trusted third-party CA.

Self-signed certificates are valuable for creating secure communication channels for internal networks when you control the environment. They offer quick deployment and cost savings and are ideal for testing, local development, or internal applications. Understanding these concepts is critical to implementing security for IP devices in general. For the Building Automation world based on BACnet, they provide the foundational knowledge for successful and robust implementation of BACnet/SC.

United States
Contemporary Control Systems, Inc.
Tel: +1 630 963 7070
Fax:+1 630 963 0109
info@ccontrols.com

China
Contemporary Controls (Suzhou) Co. Ltd
Tel: +86 512 68095866
Fax: +86 512 68093760
info@ccontrols.com.cn

United Kingdom
Contemporary Controls Ltd
Tel: +44 (0)24 7641 3786
Fax:+44 (0)24 7641 3923
ccl.info@ccontrols.com

Germany
Contemporary Controls GmbH
Tel: +49 341 520359 0
Fax: +49 341 520359 16
ccg.info@ccontrols.com

www.ccontrols.com