Creating/Using SSL Certificates

PxPlus 2017 (v14) & PxPlus 2018 (v15)

DireXions 2018

Creating/Using SSL Certificates

Agenda

- Overview of SSL
  - What it provides
  - How it is used
- What is a certificate?
- How certificates are used
- Obtaining a certificate
  - Getting a “Trusted” certificate
  - Self Signed certificates
- PxPlus options
  - Certificate validation
  - Controlling security requirements

Overview of SSL

- What does SSL stand for:

  Secure Socket Layer

  - A socket is the technical term for a network connection between machines
  - SSL is a layer between the TCP/IP interface and your application

- TLS is the new terminology

  Transport Layer Security

  - Removes the reference to 'Socket'
  - Can (*in theory*) be used on any communications

Overview of SSL

That's
Padding Oracle
On Downgraded
Legacy Encryption
NOT ME

- Versions of SSL/TLS
  - SSL 1.0 – Was never made public – Don’t Use
  - SSL 2.0 – First mainstream spec – Obsolete '95
  - SSL 3.0 – Secure except from POODLE attacks '96
  - TLS1.0 – Redesign but similar to SSL 3.0 '99
  - TLS 1.1 – Reasonable protection '06
  - TLS 1.2 – Current Standard '08
  - TLS 1.3 – Spec finalized as of March 2018
- PxPlus will connect with SSL 2.0 and above
  - This can be controlled

What SSL/TLS Provides

- Three main services that SSL/TLS provides:
  - Data encryption
  - Authentication of the server to the client
  - Authentication of the client to the server

Primarily only the first two are used

Data Encryption

- SSL/TLS encryption algorithms (ciphers)
  - Many ciphers available providing different degrees of security

Method	Description
aes	**Advanced Encryption Standard** (AES), also known as Rijndael, adopted as an encryption standard by the US government.
bf	**Blowfish** is a keyed, symmetric block cipher, designed in 1993 by Bruce Schneier
cast, cast5	**CAST** is a block cipher used in a number of products, notably as the default cipher in some versions of GPG and PGP. It has also been approved for Canadian government.
des, **des3**	The **Data Encryption Standard** (DES) is an official Federal Information Processing Standard (FIPS) for the United States in 1976, and has widespread use internationally. Triple DES (**des3**) is formed from the Data Encryption Standard (DES) cipher by using it three times
desx	**DES-X** is a variant on the DES (Data Encryption Standard) intended to increase the complexity of a brute force attack using a technique called key whitening.
Rc2	RC2 is a block cipher designed by Ron Rivest in 1987.
Rc4	RC4 (also known as ARC4 or ARCFOUR) is the most widely used software stream cipher and was often used in Secure Sockets Layer (SSL).

  - Many older ones found to be ‘crackable’
  - Some are consider completely unsafe

Data Encryption

- Ciphers provide “reversible” encryption
  - Data encrypted by ‘**Encryption key**” can only be decrypted by “**Decryption key**”
    - Key size and the algorithm determines how secure data is
    - Typical key sizes range from 128 to 4096 bits
      - 32 bit is over 4 billion thus 4096 is quite large
    - Algorithms can be found to be faulty and “Leak” answers

No cipher is 100% safe, all can be cracked given enough resources and time

Data Encryption

How are keys used:

- To send data securely to the host
  - Encryption key is made **PUBLIC**
    - Key is used to encrypt data
    - Based on the **PRIVATE** key
  - Decryption key is kept **PRIVATE** on host
    - Never should be revealed

Data Encryption

- Which cipher is used?
  - Server and client negotiate which they support
    - Client identifies which ciphers it supports
      - Supplied in order of preference
    - Server identifies which it wants
    - Client conforms
  - Server will reject any it consider unsafe or unsupported
    - Connection fails, if none are acceptable

Using insecure ciphers
will result in
PIC Compliance failure

Authentication

- Validation/Authentication of system done using certificates (X509)
  - Certificate contains the following:
    - Server name/Address
    - Start/End Date certificate is valid
    - Issuer identification
      - Name, Country, City, State/Province
    - Public Key
  - Certificates exchanged during negotiation
  - **Should** be validated for secure connection

Authentication

- What is generally validated
  - Start and end dates for certificate
  - Server name/address matches
  - Who issued/created certificate

- Optional test
  - Match to previously known public key

Multi-domain hosts with SSL use SNI
'Server Name Indication'

2017

Authentication

- **Normally** only server provides certificate
    - Client only provides a public key

- When would client require certificate?
    - SSH connections to by-pass User Id/Password
        - Provides easy secure logon of workstations with certificate
        - Also SFTP (which actually uses SSH)
    - Controlled access to specific pre-cleared clients
        - Cannot connect unless you have a known certificate

Authentication

- **But can it be trusted?**

You might not be connected to the server you think.

Just because it says its “mybank.com” doesn’t mean it is.

How to Establish TRUST

- SSL/TLS provides a mechanism that establishes ‘TRUST’
  - There are KNOWN ‘Trusted’ companies that provide ‘certificates’
    - Known as ‘Certificate Authorities’ (**CA**)
    - Managed by a consortium of large multi-nationals such as:
      - Comodo
      - Symantex
      - GoDaddy
      - Verisign
      - DigiCert

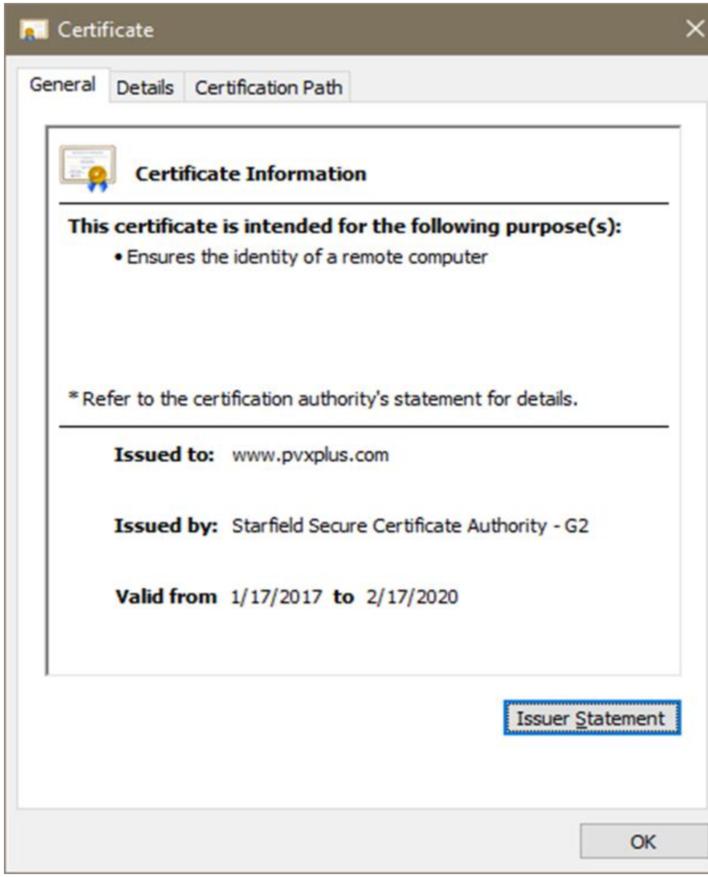
How to Establish TRUST

- CA’s have known certificates
    - Supplied with most browsers
        - Also with PxPlus 2017
    - CA’s can certify intermediate CA’s

- Server certificate includes list of who ‘certified’ the certificate
    - Compares certificate hash keys with trusted supplier tree

How to Establish TRUST

- CA’s have known certificates
  - Supplied with most browsers
    - Also with PxPlus 2017
  - CA’s can certify intermediate CA’s
- Server certificate includes list of who ‘certified’ the certificate
  - Compares certificate hash keys with trusted supplier tree



Certificate Information

This certificate is intended for the following purpose(s):

- Ensures the identity of a remote computer

* Refer to the certification authority's statement for details.

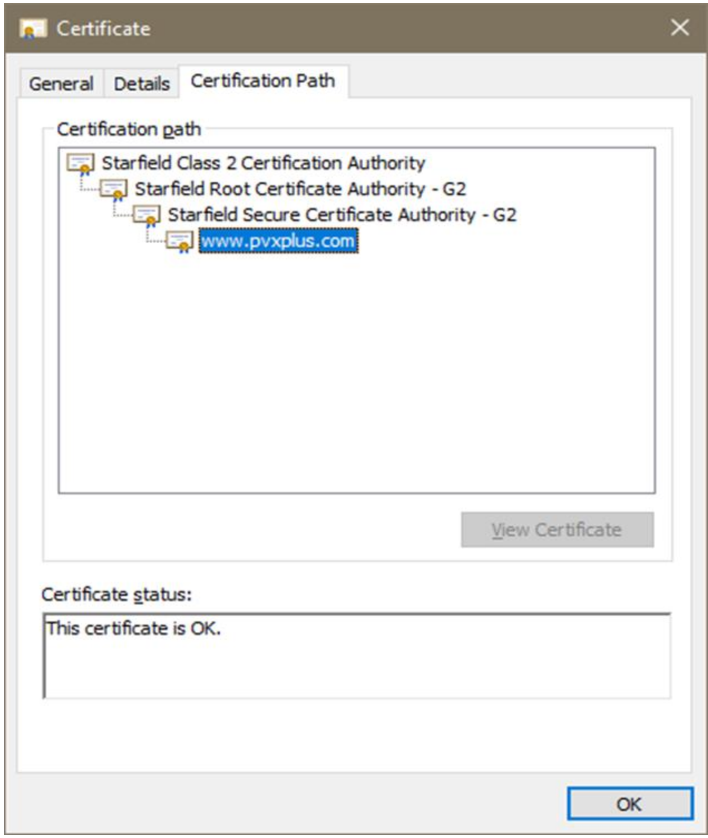
Issued to: www.pvxplus.com

Issued by: Starfield Secure Certificate Authority - G2

Valid from 1/17/2017 **to** 2/17/2020

How to Establish TRUST

- CA's have known certificates
  - Supplied with most browsers
    - Also with PxPlus 2017
  - CA's can certify intermediate CA's



Certification path

Starfield Class 2 Certification Authority
Starfield Root Certificate Authority - G2
Starfield Secure Certificate Authority - G2
www.pvxplus.com

Certificate status:

This certificate is OK.

- Server certificate includes list of who 'certified' the certificate
  - Compares certificate hash keys with trusted supplier tree

How to Establish TRUST

- Why intermediate authorities?
  - In case one gets compromised
    - It has happened and its **BAD**
  - Allows different jurisdictional zones for same company
    - Often need if rules differ
      - E.g. Rules in Europe often differ than the US
  - Shares the load of management

Getting a “Trusted” Certificate

- You need a certificate from a CA for HTTPS
    - If not trusted, browsers will complain
        - Expired certificate is the most common
        - Most will reject connection
        - Certificate MUST match site name

- How to obtain a certificate
    - Contact a CA provider
    - Costs around $100+ per year
    - Requires company background check
    - Option to purchase up to 3 years

Getting a “Trusted” Certificate

- You will need to complete a CSR

Certificate Signing Request

- A CSR contains
  - Company Name and location
    - Country, State, City
  - Department
  - Host site name
    - Can be for multiple sites – **Costs more**
  - Private key will be generated – **KEEP IT SAFE**
    - Public key forwarded with request

Getting a “Trusted” Certificate

- This is what you get back for you money:

```
-----BEGIN CERTIFICATE-----
MIIFWjCCBEKgAwIBAgIIQ9eruX5SLTkwDQYJKoZIhvcNAQELBQAwgcYxCzAJBgNV
BAYTAlVTMRAwDgYDVQQIEwdBcml6b25hMRMwEQYDVQQHEwpTY290dHNkYWxlMSUw
IwYDVQQKExxTdGFyZmllbGQgVGVjaG5vbG9naWVzLCBJbmMuMTMwMQYDVQQLEypo
dHRwOi8vY2VydHMuc3RhcmZpZWxkdGVjaC5jb20vcmVwb3NpdG9yeS8xNDAyBgNV
BAMTK1N0YXJmaWVsZCBTZWN1cmUgQ2VydGlmaWNhdGUgQXV0aG9yaXR5IC0gRzIw
HhcNMTcwMTE3MTUyMjAwWhcNMjAwMjE3MTQxMDExWjA9MSEwHwYDVQQLExhEb21h
aW4gQ29udHJvbCBWYWxpZGF0ZWQxGDAWBgNVBAMTD3d3dy5wdnhwbHVzLmNvbTCC
ASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANVri/GKVPSBJweZZLuTd0Vl
…
VLLHInOd8YzNkV0tuK4SNHAeskbTMOMHpL3lk9Vn4lE5XmD6olwwD2WVPb86wV3V
VISCDrd0pv0vQOar2vpWcygshrV+TfWIRX5IZ2XNADITrbIPnz4bPvcY+m+wmD3l
iYr6B+FRFSBCS79DRuA4cBUYixWC4TeJaRt3REKzlvMJ/2dYAtvHiI19R+AxVy6I
CR1Ym6eckW6+WuU2KaS69RJHYSMug9UNAvwYVr6tetk44HDJwdye4lHK5RgJHVrI
TtG08LRh5Wlm31IWU82ZDK3vFqn2rpWbhZGahCY9
-----END CERTIFICATE-----
```

- You will also get similar file with CA certificates

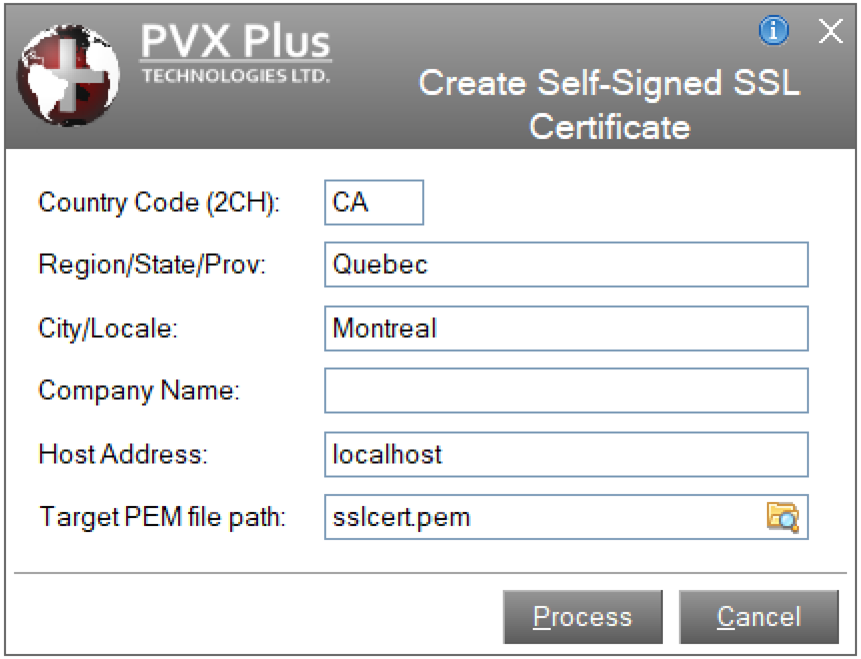
A “Self Signed” Certificate

- You can generate a certificate for yourself
  - By default, it will not be trusted
  - Can be used by all SSL/TLS software
    - Application can decide if TRUST is required
    - If trust required, users can add it to their local store
    - Includes all the same data as a standard certificate

2017

A “Self Signed” Certificate

PxPlus 2017 includes utility to create self-signed certificate



Country Code (2CH): CA

Region/State/Prov: Quebec

City/Locale: Montreal

Company Name:

Host Address: localhost

Target PEM file path: sslcert.pem

- To generate a file:

Run “*tools/sslcert”

- Uses Internet to create certificate on our servers
- Return single PEM file with certificate and key information
- Generates 2048 key
- Text mode version also available

DireXions 2018

PxPlus SSL Options

2017

Default set using PVX_CERTIFICATES environment variable

- Certificate Validation

Certificates= Ignore | Validate | Trust

- **Ignore** doesn't validate certificate (default)
- **Validate** makes sure certificate:
  - Not expired
  - Is for proper server by matching name
- **Trust** extends Validation
  - Certificate must have come from trusted CA
  - PxPlus ships with list of trusted certificates

    <pxplus exe directory>/ca-bundle.crt
  - This file must be updated periodically

Can be changed using PVX_CERTSTORE environment variable

PxPlus SSL Options

- Defining acceptable/supported Ciphers

Ciphers= *list of accepted cipher*

- Contents of list defined at www.openssl.org
- Simplest form for PCI compliance ***(currently)***

Ciphers=HIGH:MEDIUM:!ADH

- Includes ciphers with 128 bit keys or better
- Excludes DH cipher suite (Diffie Hellman)

PxPlus SSL Options

- Defining supported Protocol:

To suppress any of these protocols:

NoSSLv2, NoSSLv3, NoTLSv1, NoTLSv1.1, NoTLSv1.2

To force one specific protocol:

TLS, TLS1.1, TLS1.2

- Default will connect using any protocol from SSL v2 through TLS 1.2
  - TLS1.2 included as of PxPlus 2017

2017

PxPlus Client Server and SSL

2017

Host side CS options
(server)

Default options can be set in:
PXP_CS_OPT
Environment variable

Client side CS options
(workstation)

Default options can be set in:
PXP_CS_OPT_CLIENT
Environment variable

Future Considerations

- SSL is constantly changing to address new vulnerabilities
  - Maintain your PxPlus version current
    - We update SSL to latest options with each release
  - On Linux, keep your openssl current
  - For Windows, we ship current openssl libraries
  - If using **trust** relationships, update ca-bundle.crt from:

https://raw.githubusercontent.com/bagder/ca-bundle/master/ca-bundle.crt

Additional Resources

The help link(s) below refer to the current on-line help pages. The functionality may have been further updated since the PxPlus 2018 (version 15) release.

- SSL/TLS Certificates
- SSL Certificate Generator
- EZ Web Server
- Install Windows Services
- Let's Encrypt SSL/TLS Certificates
- Environment Variables