NetBurner

Factory Application Certificates and Keys

Products: SB700EX, SB70LC

Contents

1 Overview

This guide will use the term NetBurner Device, abbreviated as “NBD”, to refer to the SB700EX and SB70LC NetBurner devices. The factory program for these devices have similar functionality.

The NetBurner NBD Factory Application supports the following types of encrypted connections:

- SSL Web Server. A web browser may use the HTTPS to connect to the NBD.
- SSL incoming network connections for serial-to-Ethernet. If the SSL option is enabled in the NBD for the serial-to-Ethernet connection, an external host application may initiate a connection to the specified IP address and port number. The NBD uses the same SSL certificate and key used for the web server.
- SSL outgoing network connections for serial-to-Ethernet. The NBD may initiate an outgoing SSL connection to a SSL server.
- SSL outgoing network connections with certificate checking for serial-to-Ethernet. In addition to initiating the outgoing connection, the NBD will check the destination SSL Server’s certificate against a list of Certificate Authorities.
- SSH incoming network connection for serial-to-Ethernet.

2 Certificates and Keys

Description	Used By
SSL Certificate and Public Key RSA Public/Private Key Pair NBD Configuration web page: HTTPS	Web Server (HTTPS) SSL Server for incoming SSL connections
SSH RSA Public/Private Key Pair SSH DSA Public/Private Key Pair NBD Configuration web page: SSH	SSH Server for incoming SSH connections
CA Certificates NBD Configuration web page: CA Certs	SSL Client for outgoing SSL connections. If no CA Certs are defined, then no CA Certificate checking is done for outgoing SSL connections. If one or more CA Certs are defined, all outgoing SSL Client connections will perform client side certificate checking of the destination server’s SSL certificate. Note that the CA must be the same CA as the one used to create the server’s certificate.

2.1 What is in a Certificate?

A certificate contains the following information:

- Public Key
- Name
- Signature, signed by a Certificate Authority (CA)

Please see the “Creating a Code Module for SSL Client Certificates” section of the “NetBurner Security Libraries” document for information on creating a certificate.

The example below is the certificate output created by the certificate creation utility:

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
        Signature Algorithm: md5WithRSAEncryption
        Issuer: C=US, ST=California, L=San Diego, O=NetBurner, Inc.,
CN=NetBurner/emailAddress=sales@netburner.com
        Validity
            Not Before: Aug 27 17:10:41 2008 GMT
            Not After : Aug 25 17:10:41 2018 GMT
        Subject: C=US, ST=California, L=San Diego, O=NetBurner, Inc.,
CN=NetBurner/emailAddress=sales@netburner.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (512 bit)
                Modulus (512 bit):
                    00:ee:10:bd:b8:41:fb:06:ff:c9:8f:65:99:54:86:
                    2a:c5:28:6d:9d:bd:bb:4e:cc:d5:ee:8f:a4:30:98:
                    01:37:be:38:12:be:65:d4:63:75:2c:3a:43:ba:e8:
                    8d:de:e4:f0:75:59:eb:c2:3f:13:08:b0:10:78:88:
                    cb:13:a0:c7:51
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
            50:A7:32:D3:0F:49:05:E4:5C:80:BB:C8:88:05:5E:EE:93:68:CA:5B
            X509v3 Authority Key Identifier:
            keyid:50:A7:32:D3:0F:49:05:E4:5C:80:BB:C8:88:05:5E:EE:93:68:CA:5B
            DirName:/C=US/ST=California/L=San Diego/O=NetBurner,
Inc./CN=NetBurner/emailAddress=sales@netburner.com
            serial:00

            X509v3 Basic Constraints:
            CA:TRUE
    Signature Algorithm: md5WithRSAEncryption
        2d:e6:85:c4:e3:95:a4:56:41:91:74:7d:25:b9:02:ef:41:2a:
        e4:2a:c4:be:31:fd:df:38:0f:37:8b:b4:7d:d7:a0:0c:c0:bd:
        89:72:0a:1e:39:d4:5c:8c:a2:4a:1d:f1:1a:b2:59:3e:23:f0:
        d9:b1:c9:ad:9f:3c:a9:7e:15:19
```

PEM encoded certificate file usually created with the extension of “.crt” example:

```
-----BEGIN CERTIFICATE-----
MIICTjCCAfigAwIBAgIBADANBgkqhkiG9w0BAQQFADBWMQswCQYDVQQGEwJVUzEL
MAkGA1UECBMCQ0ExEjAQBgNVBAcTCVNBTiBESUVHTzESMBAGA1UEChMJTkVUQlVS
TkVSMRIwEAYDVQQDEwlORVRCVVJORVIwHhcNMTAwNDIyMjIzNDA2WhcNMjAwNDE5
MjIzNDA2WjBWMQswCQYDVQQGEwJVUzELMAkGA1UECBMCQ0ExEjAQBgNVBAcTCVNB
TiBESUVHTzESMBAGA1UEChMJTkVUQlVSTkVSMRIwEAYDVQQDEwlORVRCVVJORVIw
XDANBgkqhkiG9w0BAQEFAANLADBIAkEAvJVGJ9MpVPy9GAs14I1ixhxmrsF9gK7W
wfOTgzXPVTUKe9Gi0J5ATNU2a7HCgXnmZVjypoVzJmTq/+1ovlFz1QIDAQABo4Gw
MIGtMB0GA1UdDgQWBBRVaZ/WaGJCJwb/GxInSqiMBQGzJzB+BgNVHSMEdzB1gBRV
aZ/WaGJCJwb/GxInSqiMBQGzJ6FapFgwVjELMAkGA1UEBhMCVVMxCzAJBgNVBAgT
AkNBMRIwEAYDVQQHEwlTQU4gREVFR08xEjAQBgNVBAoTCU5FVEJVUk5FUjESMBAG
A1UEAxMJTkVUQlVSTkVSggEAMAwGA1UdEwQFMAMBAf8wDQYJKoZIhvcNAQEEBQAD
QQAgHJADyS+zhqWoANUsF0M+1XcMvpo6AiNk6Qhyl67rO4rUQ6oNbZrFjkKT/Ej5
5nSuUQS6486RY6znIAm+5daw
-----END CERTIFICATE-----
```

PEM encoded public/private key file usually created with extension “.key” example.

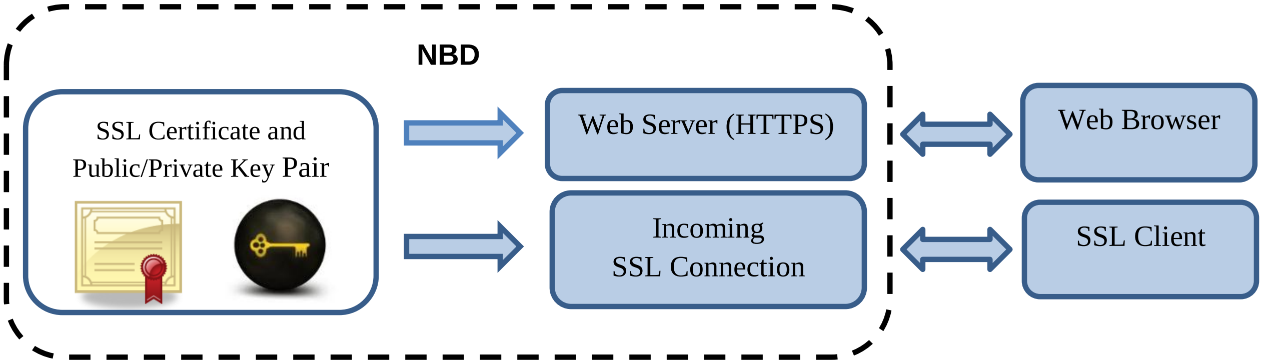
```
-----BEGIN RSA PRIVATE KEY-----
MIIBOgIBAAJBAOrfRkFnPMI0K41ufL1HLzlpf2yieGLSGE8kL2OQjX0Pp4Qq+91F
DRYD1YuKiPfjxsAkVBqlY7v23ZvzEfNcgDUCAwEAAQJAaFT2KGdrnfj+v7ysvIe6
eo5ahC9Hut4I3l78jgXQVBSeMhatb+RMyuSshgGq3+2ph6EQQABBstvuWwl5AAkU
oQIhAPtpCjpqiAQtqo1u64T/Pr5fX2IuzmbOhIvW8czDdKF3AiEA7yjxoEGMl+8o
4v8pLFZqR0s4P4G/wgScuqtCPLtjtrMCICrH5QWruxl669rFVS58gKDEeearMFQu
MD/bg6nkWKRhAiBTMuwz8vnFFUclCN069mkMmkdcGHgsN8yKR+/IDuyWbwIhAKZ9
KgZz3UZCnWHDXaelDFJI+Xdstx5XwBdTAlqwOU+L
-----END RSA PRIVATE KEY-----
```

3 SSL Certificates and Keys

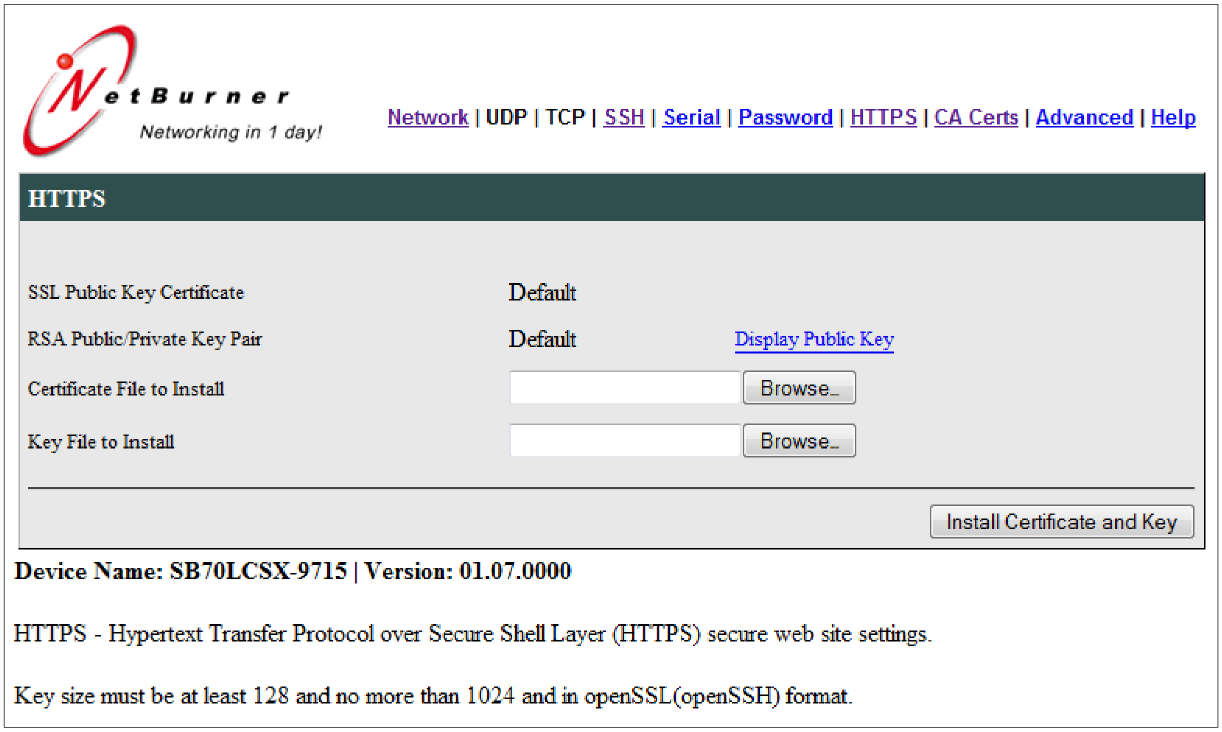
3.1 NetBurner Web Server and SSL Serial-to-Ethernet Server

The NBD Factory Application provides a default SSL certificate, which contains the public key, and a default RSA public/private key pair. The default certificate is present for the sole purpose of enabling you to communicate to the device and upload your own certificate. Creating keys is beyond the scope of this document, but you can use a utility such as OpenSSL. Certificates and keys can be purchased from companies such as Verisign.

The default and user installed SSL certs and keys are stored in the NBD flash memory, and are used for incoming SSL connections to either the web server or SSL enabled serial-to-Ethernet ports.



Certificate and key files typically have a file name suffix of .crt and .key. Once you have these files, they can be uploaded to the NBD through the HTTPS web page as shown below. Once your certificate and keys have been uploaded, the description will change from “Default” to “User Installed”.


NetBurner
Networking in 1 day!
Network | UDP | TCP | SSH | Serial | Password | HTTPS | CA Certs | Advanced | Help
Browse...
Browse...
Install Certificate and Key
Display Public Key

HTTPS

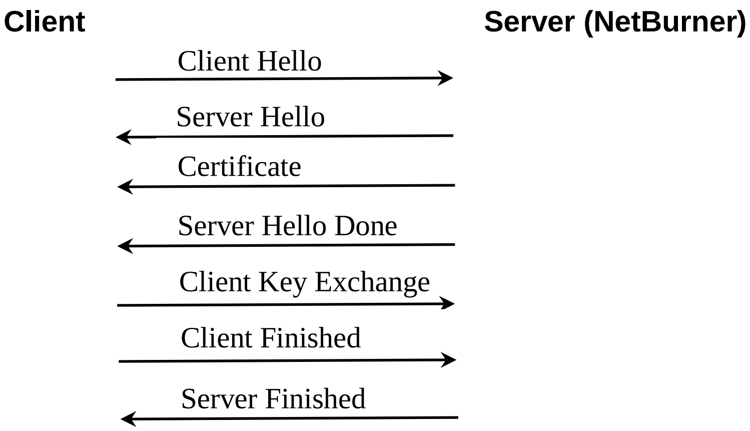
SSL Public Key Certificate	Default	
RSA Public/Private Key Pair	Default	Display Public Key
Certificate File to Install		Browse...
Key File to Install		Browse...

Device Name: SB70LCSX-9715 | Version: 01.07.0000

HTTPS - Hypertext Transfer Protocol over Secure Shell Layer (HTTPS) secure web site settings.

Key size must be at least 128 and no more than 1024 and in openSSL(openSSH) format.

3.1.1 SSL Server Handshake Sequence



3.1.2 Advanced Information for Software Developers

The default SSL certificate and keys for the factory application are defined in the program files permanentcert.h and permanentkey.h, which are compiled into the application. During the initial NBD boot sequence, this information is used to create the actual default certificate and key files stored in the Embedded Flash File System (EFFS). When a user installs their own certificate and key the default files will be overwritten in the EFFS, and become the active certificate and keys.

During normal operation the certificate and key are copied from the EFFS to a structure in memory during the boot sequence: gSslCertificatePemEncoded and gSslCertificateKeyPemEncoded.

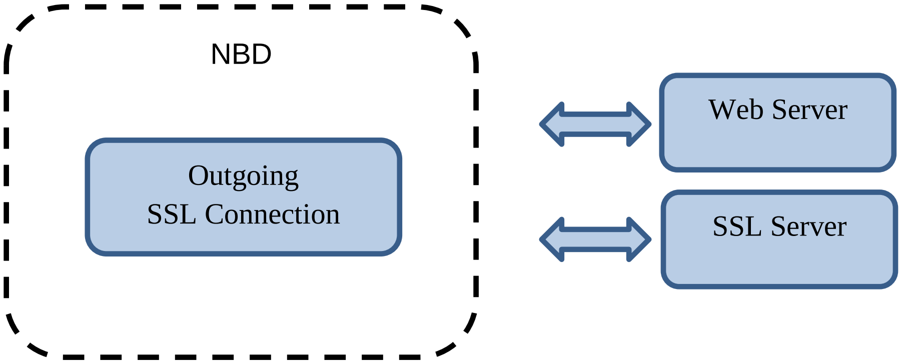
A common way to create your own certificates and keys is the OpenSSL program. If you use the NetBurner OpenSSL.exe program in the \nburn\pcbin directory, it will create a file named key.cpp in addition to the .crt and .key files. The .cpp file contains the certificate, public key and private key, and can be used to compile a custom certificate into your custom application.

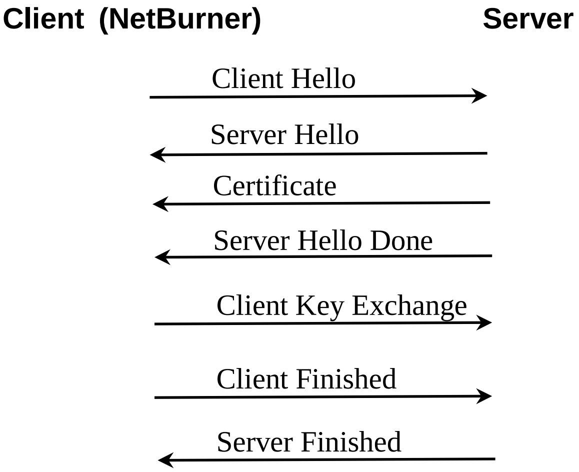
3.2 NetBurner SSL Serial-to-Ethernet Client

Client mode means that instead of the NBD listening for an incoming connection, it makes an outgoing connection on power-up or when serial data is available. Two SSL Client modes are supported:

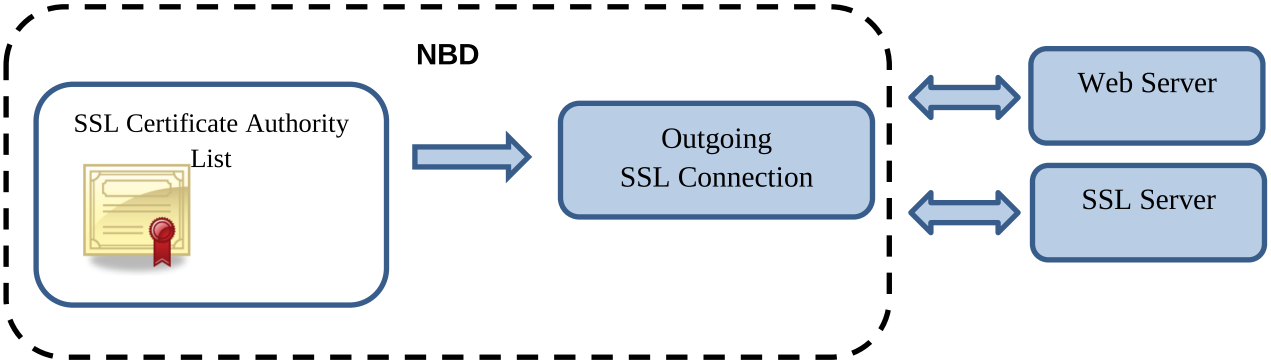
- Standard SSL Client connection.
- SSL Client connection with Certificate Authority (CA) certificate authentication of the server's certificate.

3.2.1 SSL Client Handshake Sequence Without Certificate Checking

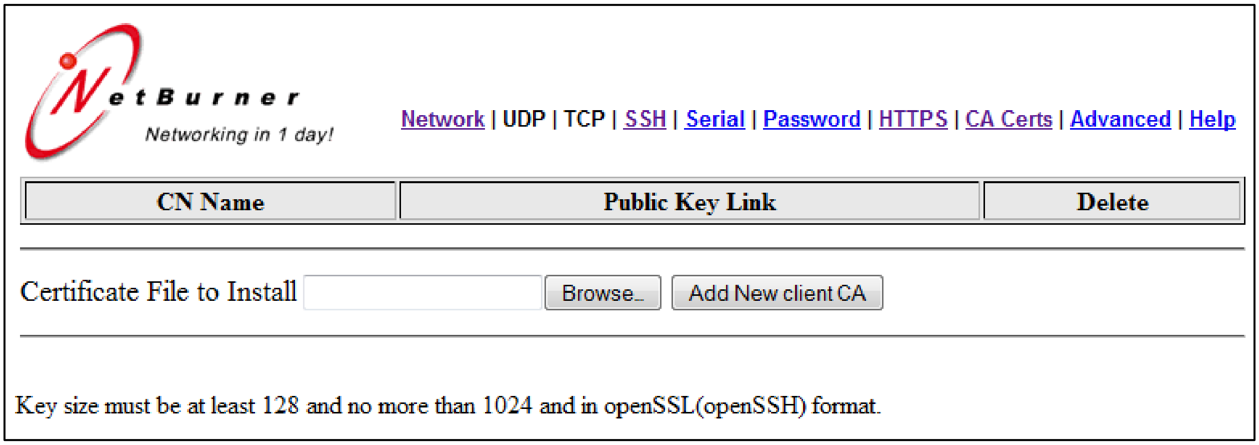




3.2.2 SSL Client Handshake Sequence with List of Certificate Authorities



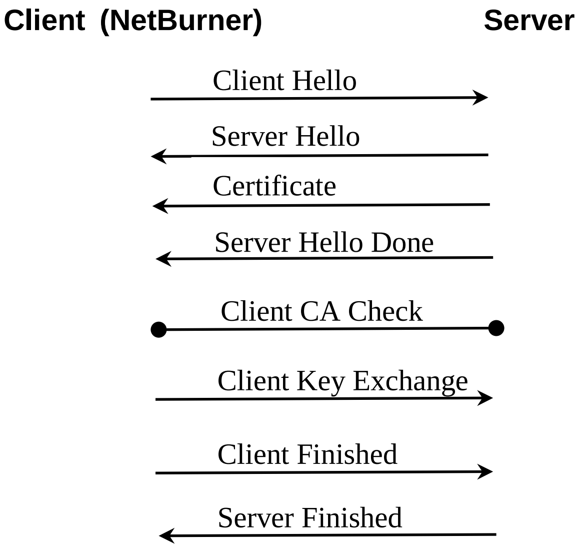
In this mode of operation the NBD, as the SSL Client, will only allow a connection if the certificate sent by the SSL Server matches a Certificate Authority certificate, which can be uploaded via the CA Certs configuration web page:


NetBurner
Networking in 1 day!
Network | UDP | TCP | SSH | Serial | Password | HTTPS | CA Certs | Advanced | Help
Browse...
Add New client CA

CN Name	Public Key Link	Delete

Certificate File to Install

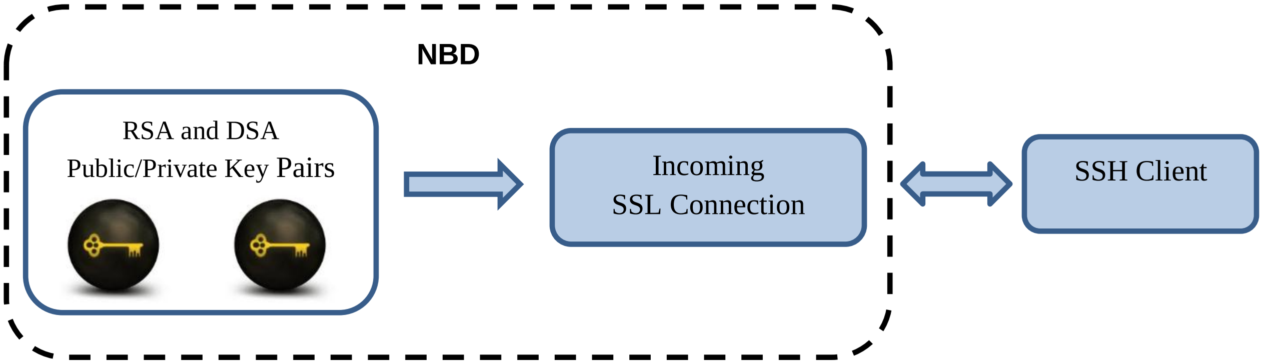
Key size must be at least 128 and no more than 1024 and in openSSL(openSSH) format.



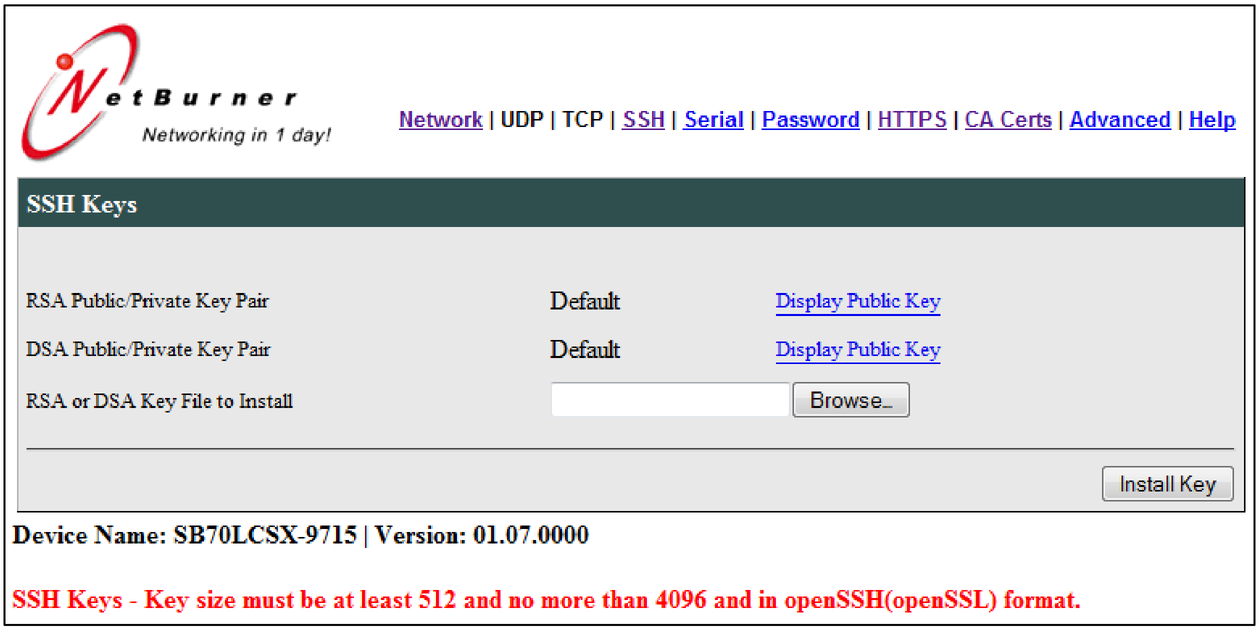
4 SSH Keys

The NBD Factory Application supports SSH Server operation and provides default RSA and DSA public and private SSH key pairs. To provide secure communication you should create or purchase your own keys. Creating keys is beyond the scope of this document, but you can use a utility such as OpenSSL.

The default and user installed SSH keys are stored in the NBD flash memory, and are used for incoming SSH connections to SSH enabled serial-to-Ethernet ports.



Key files can be uploaded to the NBD through the SSH Keys web page as shown below. Once your keys have been uploaded, the description will change from “Default” to “User Installed”.


NetBurner
Networking in 1 day!
Network | UDP | TCP | SSH | Serial | Password | HTTPS | CA Certs | Advanced | Help
Browse...
Install Key

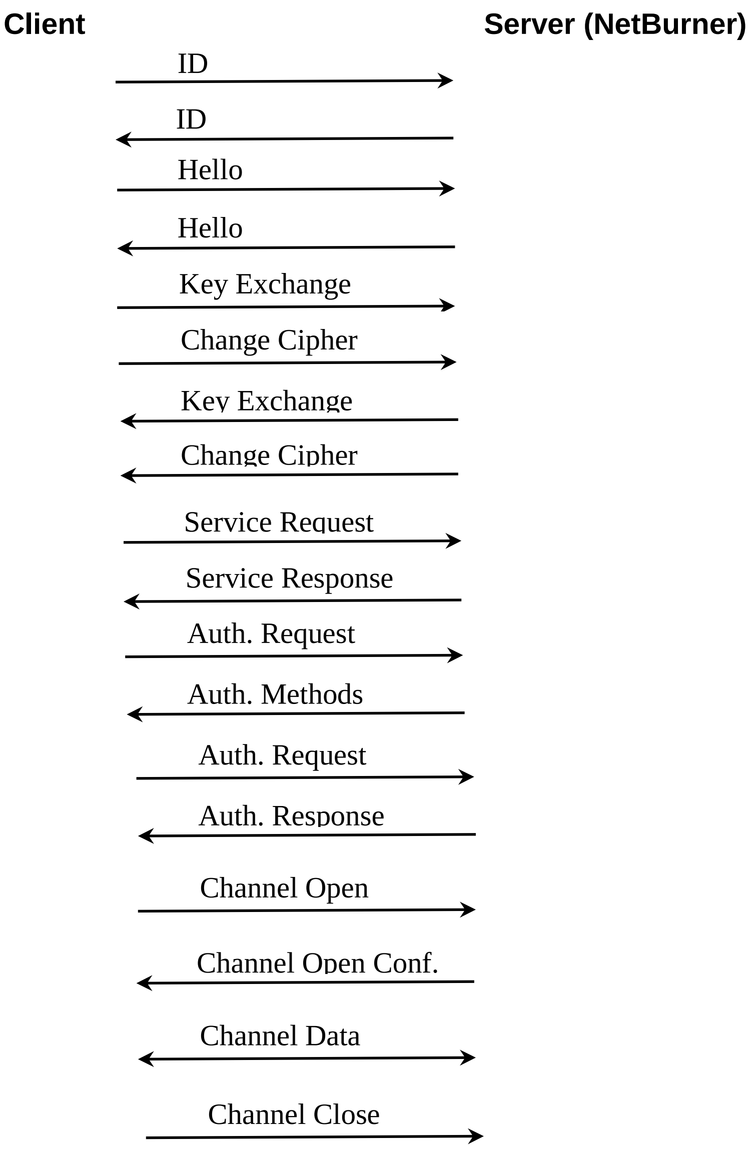
SSH Keys

RSA Public/Private Key Pair	Default	Display Public Key
DSA Public/Private Key Pair	Default	Display Public Key
RSA or DSA Key File to Install		

Device Name: SB70LCSX-9715 | Version: 01.07.0000

SSH Keys - Key size must be at least 512 and no more than 4096 and in openSSH(openSSL) format.

4.1.1 SSH Server Handshake Sequence



4.1.2 Advanced Information for Software Developers

The default SSH RSA and DSA keys for the factory application are defined in the program files: permannetkeyrsa.h and permanentkeydsa.h, and are built into the application at compile time. During the initial NBD boot sequence, this information is used to create the actual default key files stored in the Embedded Flash File System (EFFS). When a user installs their own keys, the default files will be overwritten in the EFFS and become the active certificate and keys.

During normal operation the certificate and key are copied from the EFFS to a structure in memory during the boot sequence: gSshRsaKeyPemEncoded and gSshDsaKeyPemEncoded. A call to SshConnect() cause user supplied function SshUserGetKey() to load the key from global data. Please refer to the SSH library API and example programs for more information.