

**GEORGIA SOFTWAREWORKS**

---

SSH Server for Windows Server

2019/2016/2012/R2/2008/R2 and Windows 10/8/7

*Keep it Secure – Simply*

# User's Guide

THIS PAGE INTENTIONALLY LEFT BLANK

---

GEORGIA SOFTWORKS

# SSH Server

---

Copyright © 1997-2020, Georgia SoftWorks, All Rights Reserved  
Public Square  
17 Hwy 9 South • PO Box 729  
Dawsonville Georgia 30534  
Telephone 706.265.1018 • Fax 706.265.1020  
<https://www.georgiasoftworks.com>

Copyright © Georgia SoftWorks, 1997-2018 All Rights Reserved.

User's Manual, Version 8.10.0003, Nov Aug 12, 2020

Microsoft, Windows, Windows VISTA, Windows XP, Windows 2000 Windows NT, Windows 98, Windows 95, Windows 7, Windows 8, Windows 10, Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2 , Windows Server 2008, Windows Server 2003 are trademarks of Microsoft Corporation. SAP, SAPConsole are trademarks of SAP AG. SecureCRT, F-Secure, PuTTY, PuTTYgen are trademarks of their respective companies.

**THIS PROGRAM IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.**

**LICENSOR MAKES NO WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, ORAL OR WRITTEN, REGARDING THE PROGRAM OR DOCUMENTATION AND HEREBY EXPRESSLY DISCLAIMS ALL OTHER EXPRESS OR IMPLIED WARRANTIES, INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. LICENSOR DOES NOT WARRANT THE PROGRAM WILL MEET YOUR REQUIREMENTS OR THAT ITS OPERATION WILL BE UNINTERRUPTED OR ERROR FREE.**

**IN NO EVENT WILL GEORGIA SOFTWORKS BE LIABLE TO YOU FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE SUCH PROGRAMS.**

**COPYING:**

**WHILE YOU ARE PERMITTED TO MAKE BACKUP COPIES OF THE SOFTWARE FOR YOU OWN USE AND PROTECTION, YOU ARE NOT PERMITTED TO MAKE COPIES FOR THE USE OF ANYONE ELSE.**

**LICENSE:**

**YOU ARE LICENSED TO RUN THIS SOFTWARE ON A SINGLE WINDOWS 7/8/10/VISTA/2008/R2/2012/R2/2016/2019, NT/XP/2000/2003 SYSTEM. THE GEORGIA SOFTWORKS WINDOWS 7/8/10/VISTA/2008/R2/2012/R22016/2019, NT/XP/2000/2003 SSH SERVER SOFTWARE MAY BE INSTALLED ON A SINGLE WINDOWS 7/8/10/VISTA/2008/R2/2012/R2/2016/2019, NT/XP/2000/2003 SYSTEM.**

This Page Left Intentionally Blank

# Table of Contents

<b>FEATURES AT A GLANCE .....</b>	<b>11</b>
<b>OVERVIEW .....</b>	<b>13</b>
EASE OF USE .....	16
COMPONENT ARCHITECTURE .....	17
<b>INSTALLATION .....</b>	<b>19</b>
OVERVIEW .....	19
PROCEDURE.....	20
<b>REGISTRATION.....</b>	<b>24</b>
REGISTRATION VIA SOFTWARE SERIAL NUMBER .....	24
<i>How to Register the Software</i> .....	24
FLOATING LICENSE – OVERVIEW .....	29
<i>Floating License – Hardware Key Installation Instructions</i> .....	31
<i>Uninstall Floating License – (Hardware Key)</i> .....	34
<b>GSW SSH SERVER.....</b>	<b>35</b>
<b>GSW FIPS 140-2 COMPLIANT OPTION .....</b>	<b>37</b>
SOFTWARE REQUIREMENTS .....	37
ENABLE OPTION.....	38
<i>ENABLE FIPS 140-2 ON SSH SERVER</i> .....	38
<i>ENABLE FIPS 140-2 ON GSW MOBILE/CE and DESKTOP CLIENTS</i> .....	39
FIPS 140-2 CONNECTIONS .....	41
<b>CONFIGURATION .....</b>	<b>42</b>
REGISTRY KEY LOCATIONS.....	42
ALLOW USER TO USE SCP CHANNEL .....	42
ALLOW USER TO START A REMOTE SHELL.....	43
ALLOW USER TO USE SFTP SUBSYSTEM.....	44
SPECIFY THE SFTP AND SCP ROOT FOLDER.....	45
SPECIFY UNC PATH FOR SFTP AND SCP .....	46
ALLOW USER CERTIFICATE LOGON .....	48
ALLOW USER PUBLIC KEY LOGON.....	49
ALLOW USER NAME/PASSWORD LOGON.....	49
ALLOW USER GSSAPI LOGON.....	50
LISTEN ON IPV4 INTERFACES .....	51
LISTEN ON IPV6 INTERFACES .....	51
ALLOW USE OF THE “EXEC” CHANNEL .....	52
ENABLE RSA HOST KEY.....	53
ENABLE DSA HOST KEY.....	54
ENABLE ECDSA HOST KEY .....	54
ENCRYPTION ALGORITHM CATEGORIES AND LISTS OVERVIEW .....	56
<i>Specify Key Exchange Algorithms</i> .....	57
<i>Specify Ciphers</i> .....	58
<i>Specify Message Authentication Code Algorithms (MACs)</i> .....	59
ALGORITHM SELECTION FOR GSW DESKTOP SSH CLIENT .....	60
<i>Host Key Algorithms List</i> .....	60
<i>Key Exchange Algorithm List</i> .....	61

*Ciphers List* .....62  
*Message Authentication Code (MACs) list*.....63  
*SSH Desktop Client Command line Syntax*.....64  
CHANGE THE SSH PORT NUMBER.....65  
LOCATION OF SSH SERVER RSA PRIVATE KEY.....66  
LOCATION OF SSH SERVER DSA PRIVATE KEY.....67  
LOCATION OF SSH SERVER ECDSA PRIVATE KEY.....68  
LOCATION OF FINGERPRINTS FOR ALL HOST KEYS .....69

**SSH SERVER MAPPING TOOL FOR CERTIFICATES AND PUBLIC KEYS .....72**  
CERTIFICATE ONE-TO-ONE MAPPING.....73  
CERTIFICATE MANY-TO-ONE MAPPING .....73  
PUBLIC KEY 1-TO-1 MAPPING.....74  
CERTIFICATION VALIDATION – CERTIFICATE TRUST LIST.....75

**PUBLIC/PRIVATE KEY CREATION AND USE.....76**  
PUBLIC / PRIVATE KEY INTRODUCTION.....76  
CREATING A PUBLIC / PRIVATE KEY .....77  
INSTALL PUBLIC KEY ON THE SERVER AND MAP TO WINDOWS USER ACCOUNT.....80  
INSTALL AND CONFIGURE PRIVATE KEY ON THE CLIENT.....83

**SSH CLIENTS .....93**  
GSW ANDROID SSH CLIENTS.....93  
GSW WINDOWS SSH CLIENTS .....93  
*GSW DESKTOP CLIENT*.....94  
*Windows Mobile Clients*.....95  
*Windows Mobile*.....95  
*Windows CE 4.2+ Devices* .....97  
THIRD PARTY SSH CLIENTS.....100  
*Specify Domain with a 3<sup>rd</sup> Party Client* .....103

**REGISTRY VARIABLES .....104**

**SSH ALGORITHMS SUPPORT .....106**

**KEY EXCHANGE ALGORITHMS.....106**  
**HOST KEY ALGORITHMS**.....106  
**KEY EXCHANGE ALGORITHMS** .....106

**CIPHERS .....107**  
**CIPHERS** .....107  
**MACS**.....108

**HOST KEY TYPES .....108**  
**PUBLIC KEY ALGORITHMS** .....108

**COMPRESSION .....109**  
**COMPRESSION**.....109

**GSW SSH SERVER SUBSCRIPTION .....111**  
HOW TO UPDATE THE SOFTWARE .....112  
HOW TO RENEW THE GSW SUBSCRIPTION .....112

**SSH SERVER FOLDER LAYOUT.....113**

**SYSTEM SIGNATURE - IMPORTANT PLEASE READ .....115**

**TECHNICAL SUPPORT .....116**

    PROVIDE LOG FILES TO GSW TECHNICAL SUPPORT.....116



**TABLE OF FIGURES**

Figure 1: GSW Server Products Block Diagram .....17

Figure 2: GSW Telnet Server Block Diagram .....18

Figure 3: GSW SSH Server Block Diagram.....18

Figure 4: Installation Welcome Screen.....20

Figure 5: Installation – Choose Destination Folder .....21

Figure 6: Installation – Command Shell Status Lines .....22

Figure 7: Installation Complete .....22

Figure 8: GSW UTS Program Group.....23

Figure 9: SSH Installation Status (Your version will be shown in the fields above) .....23

Figure 10: Registration – SSH Shield is not registered for use .....24

Figure 11: GSW Registration - Initial Screen.....26

Figure 12: Registration - Serial Number Applied.....27

Figure 13: Registration Successful Screen.....28

Figure 14: Registration Verification.....28

Figure 15: Registration - Verify that FIPS 140-2 is Enabled .....29

Figure 16: Floating License – Parallel Port .....30

Figure 17: Floating License - USB Port .....30

Figure 18: Floating License - Hardware Key .....30

Figure 19: Hasp Preparing to Install.....31

Figure 20: Sentinel welcome screen.....32

Figure 21: SafeNet License Agreement.....32

Figure 22: gemalto Sentinel Runtime Setup .....33

Figure 23: gemaltor Sentinel Runtime Setup Progress bar .....33

Figure 24: SafeNet Validating Install .....34

Figure 25 - GSW Software Installation Status Tool .....35

Figure 26: Control Panel - GSW SSH Services Started.....36

Figure 27: GSW True FIPS 140-2 Connection – Server and Client .....38

Figure 28: FIPS 140-2 Option Enabled .....38

Figure 29: Desktop Client "-i" option issued.....39

Figure 30: Enable FIPS 140-2 on GSW Mobile Clients.....40

Figure 31: Verify FIPS 140-2 Compliant Connections .....41

Figure 32 - SSH Certificate Mapping Tool.....72

Figure 33 - One-to-one certificate mapping .....73

Figure 34 - Many-to-one certificate mapping.....74

Figure 35 - Public Key Mappings - 1-to-1 .....74

Figure 36 - Certificate Validation Certificate Trust List .....75

Figure 37: Puttygen Tool for Creating Public/Private Key Pair.....77

Figure 38: Puttygen Tool Uses Mouse-Movement to Create Randomness for Key Creation .....78

Figure 39: Puttygen with Generated Key .....79

Figure 40: Certificate Mapping Tool .....80

Figure 41: Mapping an Account to a Public Key.....81

Figure 42: Certificate Mapping Tool – Public Key Mappings .....81

Figure 43: Certificate Mapping Tool Dialogue Box .....82

Figure 44: Opening the Client on a Device to Set-up Public Key Log-on .....83

Figure 45: Selecting a Connection profile.....84

Figure 46: Selecting Options on GSW SSH2 Client to Configure Public Key Log-on .....85

Figure 47: Selecting Public Key Log-on .....86

Figure 48: Configuring Public Key Log-on.....87

Figure 49: Configuring the Client by Providing a Private Key .....88

Figure 50: Browsing to Find Private Key .....89

Figure 51: Private Key Imported into Client .....90

Figure 52: Client Launch Using Public / Private Key Authentication .....91

Figure 53: Session Connected via Public / Private Key Authentication .....92

Figure 54: GSW SSH Desktop Client.....94

Figure 55: GSW PPC 2003 Client .....95

Figure 56: GSW PPC 2003 Client – Options .....96

Figure 57: GSW PPC 2003 Client - SAPConsole - SSH .....96

Figure 58: Psion-Teklogix Initial Screen .....97

Figure 59: Psion-Teklogix – Session Menu Items.....97

Figure 60: Psion-Teklogix Connection Settings .....98

Figure 61: Psion-Teklogix – Save Settings .....98

Figure 62: Psion-Teklogix running SAP via SAPConsole.....99  
 Figure 63: Psion-Teklogix Save Client Settings Menu.....99  
 Figure 64: 3rd Party Client – SecureCRT – SAPConsole ..... 100  
 Figure 65: 3rd Party Client - PuTTY - Unicode..... 101  
 Figure 66: 3rd Party Client - F-Secure SSH Client ..... 102  
 Figure 67: Installation Folder Layout of the GSW UTS..... 113  
 Figure 68: Installation Folder Layout of the GSW SSH Shield ..... 114

## Table of Tables

Table 1: GSW Software versions required for FIPS 140-2.....37  
 Table 2: Device Operating System Versions Required for FIPS 140-2.....37  
 Table 3: GSW Desktop Host Key Algorithms.....60  
 Table 4: GSW Desktop Key Exchange Algorithms .....61  
 Table 5: GSW Desktop Ciphers - SSH and with FIPS 140-2.....62  
 Table 6: GSW Desktop MACs both SSH and FIPS 140-2.....63  
 Table 7: GSW SSH Client Platforms.....93  
 Table 8: SSH Host Key and Key Exchange Algorithms ..... 106  
 Table 9: SSH Ciphers ..... 107  
 Table 10: SSH HMACs ..... 108  
 Table 11: SSH Public Key Algorithms ..... 108  
 Table 12: SSH Compression..... 109  
 Table 13: FIPS 140-2 certificate links..... 110  
 Table 14: Version Upgrade Pricing **with** GSW Subscription Plan ..... 111  
 Table 15: Version Upgrade Pricing **Without** Subscription Plan..... 111  
 Table 16: Steps to Renew the GSW Subscription Plan..... 112

## Typographic Conventions

<i>Italics:</i>	are used to emphasize certain words, especially new terms or phrases when they are introduced.
<b>Initial Caps Bold:</b>	Words that appear in initial caps boldface represent menu options, buttons, icons or any object that you may click.
Courier:	This font represents anything you must type.
"<enter>"	This represents the enter key.

## Terms/Abbreviations

UTS	GSW Universal Terminal Server
SSH	Secure Shell Version 2 Always refers to SSH version 2 (SSHv2) except where noted
SSH SHIELD	This is the application and interface installer for the GSW SSHv2 Interface
SSH SHIELD Certificate Mapping Tool	This is the GSW GUI tool that is used when configuration and managing the mapping of Digital Certificates, Public Keys and CTL's. Often called the GSW Mapping Tool or GSW Certificate Mapping Tool.
Telnet Server	Unless noted otherwise this refers to the GSW UTS with the default Telnet Protocol
Windows	Refers to Microsoft Windows Operating Systems 10/8/7 and Windows Server 2008/R2/2012/R2/2016/2019 unless otherwise noted.

## Features at a Glance

Offering Secure Remote Logon, Secure Data Exchange, Secure Network Services and Secure Access to your Application on an Insecure Network.

### Georgia SoftWorks SSH Server

- Complete Data Stream Encryption  
AES-256, 'chacha20-poly1305@openssh.com', and the strongest modern ciphers supported ([see below](#))
- Easy to Install and Use  
Defaults provide strong encryption  
No Certificate provision required (*However, available if you want it*)
- Automatic Generation and installation of RSA, DSA and ECDSA Host Keys
- Host Fingerprints file holds key fingerprints for all host keys offered for server-to-client authentication.
- FIPS 140-2 Compliant Option
- IPv6 Support
- Integrated with GSW UTS feature set including GUI Configuration Tool
- Perfect Support for ALL PC Keys and International Characters
- GSW SSH Clients for Windows Desktops, PPC 2003, Windows CE .Net 4.2+, Windows Mobile (WM5)+ class devices.
- GSW companion product - GSW ConnectBot is an SSH client for Android, and the most secure commercial SSH client available.

#### Elliptic Curve Cryptography Support for

- Server-to-client authentication
- Key Exchange
- Public key authentication

<ul style="list-style-type: none"> <li>• <b>Host Key types</b> <ul style="list-style-type: none"> <li>• <code>'rsa-sha2-512'</code></li> <li>• <code>'rsa-sha2-256'</code></li> <li>• <code>'ecdsa-sha2-nistp521'</code></li> <li>• <code>'ssh-rsa'</code></li> <li>• <code>'ssh-dss'</code></li> </ul> </li> <li>• <b>Key Exchange algorithm</b> <ul style="list-style-type: none"> <li>• <code>'curve25519-sha256@libssh.org'</code></li> <li>• <code>'curve25519-sha256'</code></li> <li>• <code>'diffie-hellman-group-exchange-sha256'</code></li> <li>• <code>'diffie-hellman-group14-sha256'</code></li> <li>• <code>'diffie-hellman-group16-sha512'</code></li> <li>• <code>'diffie-hellman-group18-sha512'</code></li> <li>• <code>'ecdh-sha2-nistp521'</code></li> <li>• <code>'ecdh-sha2-nistp384'</code></li> <li>• <code>'ecdh-sha2-nistp256'</code></li> <li>• <code>'ext-info-c'</code></li> <li>• <code>'diffie-hellman-group-exchange-sha1'</code></li> <li>• <code>'diffie-hellman-group-14-sha1'</code></li> <li>• <code>'diffie-hellman-group1-sha1'</code></li> </ul> </li> <li>• <b>HMAC algorithms</b> <ul style="list-style-type: none"> <li>• <code>'hmac-sha2-512-etm@openssh.com'</code></li> <li>• <code>'hmac-sha2-256-etm@openssh.com'</code></li> <li>• <code>'hmac-sha2-512'</code></li> <li>• <code>'hmac-sha2-256'</code></li> <li>• <code>'hmac-sha1'</code></li> <li>• <code>'hmac-sha1-96'</code></li> <li>• <code>'hmac-md5'</code></li> <li>• <code>'hmac-sha1-etm@openssh.com'</code></li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• <b>Ciphers</b> <ul style="list-style-type: none"> <li>• <code>'chacha20-poly1305@openssh.com'</code></li> <li>• <code>'aes256-gcm@openssh.com'</code></li> <li>• <code>'aes128-gcm@openssh.com'</code></li> <li>• <code>'aes256-ctr<sup>1</sup>'</code></li> <li>• <code>'aes128-ctr<sup>2</sup>'</code></li> <li>• <code>'aes192-ctr<sup>3</sup>'</code></li> <li>• <code>'3des-ctr<sup>4</sup>'</code></li> <li>• <code>'aes256-cbc'</code></li> <li>• <code>'aes192-cbc'</code></li> <li>• <code>'aes128-cbc'</code></li> <li>• <code>'3des-cbc'</code></li> <li>• <code>'blowfish-cbc'</code></li> <li>• <code>'rijndael256-cbc'</code></li> <li>• <code>'rijndael192-cbc'</code></li> <li>• <code>'Rijndael128-cbc'</code></li> <li>• <code>'rijndael-cbc@lysator.liu.se'</code></li> <li>• <code>'cast128-cbc'</code></li> </ul> </li> </ul>
<p><i>Plus GSW Digital Certificate Based Authentication</i></p> <ul style="list-style-type: none"> <li>• Public Key Authentication with Microsoft IIS like certificate to user account mapping</li> <li>• <b>'One-to-one'</b> and <b>'Many-to-one'</b> mapping methods that also support certificate trust lists (CTL).</li> <li>• Certificate mapping tool also supports public key to user account mapping</li> <li>• Single Sign On through NTLM and Keberos over GSSAPI ('gssapi-with-mic')</li> <li>• Certificate based authentication through: <ul style="list-style-type: none"> <li>• <code>'x509v3-sign-rsa'</code> and <code>'x509-sign-dss'</code> public key authentication standards</li> <li>• Integrated with the Microsoft Certificate Stores</li> </ul> </li> </ul>	

<sup>1</sup> Ciphers - aes256-ctr when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>2</sup> Ciphers - aes128-ctr when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>3</sup> Ciphers - aes192-ctr when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>4</sup> Ciphers - 3des-ctr when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

## Overview

*The GSW Secure Shell (SSH) Server provides Secure Remote Access to your Windows Host including Secure Remote Logon, Data Exchange, and Access to your Application on an Insecure Network*

Thank you for purchasing the Georgia SoftWorks (GSW) SSH Server for Windows Server 2019/2016/2012/R2/2008/R2 and Windows 10/8/7. The GSW SSH Server provides unparalleled performance and includes the powerful features needed to achieve operational objectives in demanding commercial and industrial environments. The growing concern that sensitive data must not be available to unauthorized third parties demands that a client can securely access the remote server. This is especially important for RF access to a server.

Strong "End-to-End" encryption is employed with the GSW SSH Server. No clear text username and passwords are transmitted across the network. No clear text application data is transmitted across the network. All the data is encrypted using the strongest encryption available to provide complete confidentiality.

A Federal Information Processing Standards Publication (FIPS) 140-2 compliant option is available and may be purchased for the GSW SSH Server. This standard specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive or valuable data. This option is available to Federal agencies, including the US Military. The option is also available for purchase by other organizations such as state governments, educational and research institutions, commercial businesses and other entities with the need or desire to comply with this security requirement for cryptographic modules standard.

The GSW SSH Server is useful in a wide variety of environments that require Secure Remote Access and Strong Encryption that include:

- RF Application, Barcode Scanner, etc. (Warehousing, Inventory, Medical, etc.)
- SAP AG's SAPConsole
- HighJump, QAD and more
- Application Service Providers (ASP), Legacy Applications
- System Administration, Software Development and more!
- The [GSW Business Tunnel](#) is an excellent client application for the GSW SSH Server providing secure web browsing, email access, RDP and much more.

The GSW SSH provides SSH (SSH version 2) operation rather than the older iteration SSH1 (SSH version 1) operation. In addition to being faster, smaller and more flexible, SSH provides significant security improvements. Even though SSH1 implementations exist, they are becoming fewer and are

usually not recommended. GSW has chosen to provide the strongest, fastest and version of SSH – SSHv2.

An extremely important aspect of the GSW SSH Server is the ease of installation. Complex and lengthy security configuration has been either eliminated or reduced to a minimum in order to get your application up and running fast without forsaking performance or compromising desired security. You do not have the administrative complexity of public/private keys and certificates when using the GSW SSH Server default settings.

Secure Remote Login, Secure Access to the Application and ensuring Data Integrity are the primary areas for concern when securing an application and the GSW SSH Server is optimized to address these needs.

### **Strong Authentication**

The GSW SSH Server offers the Strongest Authentication features available for Windows.

In addition to User Name/Password Authentication, the GSW SSH Server for Windows offers Public Key Authentication with a GUI Internet Information Server (IIS) *like certificate* to user account mapping. This includes 'One-to-one' and 'Many-to-one' mapping methods and also supports certificate trusts lists (CTL). This mapping works with all user accounts including accounts defined in the Active Directory. Additionally, the GSW GUI mapping tool allows *public key* to user account mapping.

To learn more about GSW Digital Certificate Based Authentication, please visit the GSW website:

<https://www.georgiasoftware.com/feature/ssh-server-windows/server-certificate-based-authentication-x509v3>

### **Secure Remote Login**

The GSW SSH Server only allows connections from SSH clients. This ensures that all user data is encrypted prior to leaving the local client device. The data is decrypted at the remote GSW SSH Server. This includes authentication data such as the *username* and *password* that is required to login to the remote server. The encryption is transparent, and thus the user will not perceive much, if any, variance between operation of a telnet and SSH client.

The SSH connection ensures that the Login and Authentication data is encrypted so that a malicious party cannot intercept the sensitive information.

## **Secure Access to Your Application (Secure Data Exchange)**

Since the connection between the SSH client and the GSW SSH Server is encrypted, the data transmitted is not readable by unauthorized parties. When the User is authenticated, a shell is started (cmd.exe), where the user can perform remote command execution or start applications. All data transmitted between the client and the server is encrypted. No one can "snoop" the connection and intercept clear text data because none exists!

## **Data Integrity**

Data Integrity is essential for secure data exchange. The data received must be exactly the same as the data sent; otherwise an unauthorized party may have modified the data during the transmission. The SSH Transport layer ensures that the data received has not been modified from the data sent. This is accomplished by including a message authentication code (MAC) with each packet transmitted. The MAC is determined prior to encryption using the contents of the packet, a "Shared Secret" between the SSH client and SSH server and a packet sequence number.



## Ease of Use

Many of the complex and lengthy configurations issues are automatically defined by the GSW SSH Server. It has been observed that an overwhelming majority of customers do not need nor desire to set every possible option available for SSH Security.

Most customers want the strongest security that is practical to implement. Through much dialog with our resellers and customers who use RF environments a main theme emerged. The requirement to "Keep it secure – simply" was paramount.

The installation of the GSW SSH Server is very quick. You will have users connecting with the security of powerful SSH encryption much sooner than expected.

- [No Encryption Method has to be specified.](#)  
Many environments must ensure that the Windows Username and Password are encrypted as well as the data. GSW SSH Server provides *complete* confidentiality by defaulting to a very strong encryption method.

### The GSW SSH Server defaults to the following:

**Host Key Algorithms:** rsa-sha2-512,rsa-sha2-256,ecdsa-sha2-nistp521,ssh-rsa,ssh-dss

**KEX algorithms:** curve25519-sha256,curve25519-sha256@libssh.org,diffie-hellman-group18-sha512,diffie-hellman-group16-sha512,diffie-hellman-group-exchange-sha256,diffie-hellman-group14-sha256,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1

**Ciphers:** aes256-gcm@openssh.com,chacha20-poly1305@openssh.com,aes256-ctr,aes192-ctr,3des-cbc,aes128-ctr,aes128-gcm@openssh.com,aes256-cbc,rijndael128-cbc,rijndael-cbc@lysator.liu.se,aes192-cbc,rijndael192-cbc,aes128-cbc,rijndael128-cbc,cast128-cbc,blowfish-cbc

**MACs:** hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512,hmac-sha2-256,hmac-sha1-etm@openssh.com,hmac-sha1,hmac-sha1-96,hmac-md5,none

**Public Key Algorithms:** rsa-sha2-512,rsa-sha2-256,ssh-ed-25519,ssh-rsa,ssh-dss

AES-256 is the generally accepted strongest encryption standard offered by SSH – it is the Advanced Encryption Standard using a 256 bits cryptographic key. This is also known as the Rijndael algorithm which is a symmetric block cipher capable of using cipher keys that have 128, 192 and 256 bit lengths to process data blocks of 128 bits.

The GSW SSH server can be configured to refuse a connection if the SSH client can not operate with AES-256. Weaker encryptions only compromise the security of the connection so **only the strongest encryption can be configured** to ensure the strongest protection - while maintaining exceptional performance. AES-256 encryption is available on almost all SSH clients. Of course other encryptions are supported such as 3DES. The GSW SSH Server will negotiate with the client to agree on the algorithm unless configured otherwise.

- **No manual installation of certificates needed**

Additionally, it has been identified that in many cases the administrative requirements for public and private certificate installation are not needed or desired. Using traditional, manual methods the installation of certificates on RF devices can be complex and cumbersome. No public/private key generation or administration is required.

However, those with the requirements can take full advantage of the security offered by Digital Certificates and Public Keys using the innovative and easy to use SSH Shield Certificate Mapping Tool.

## Component Architecture

The GSW SSH is composed of:

- The GSW Universal Terminal Server (UTS)
- The GSW SSH Shield

The GSW UTS is the software module that contains the core software for the GSW Server products, and the majority of the Advanced Features for the GSW Server Products

## GSW SERVER PRODUCTS



Figure 1: GSW Server Products Block Diagram

The GSW UTS standard option for the Protocol and Interface is the Telnet Interface. This configuration is marketed and sold as the GSW Telnet Server.

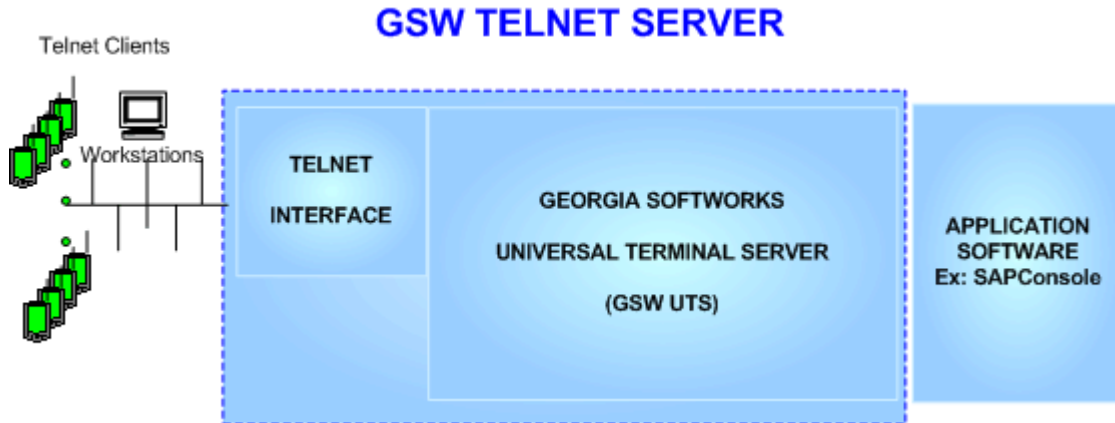


Figure 2: GSW Telnet Server Block Diagram

The GSW UTS SSH interface is installed by applying the GSW SSH Shield to the GSW UTS. The GSW SSH Shield **disconnects the Telnet Protocol Interface** and installs the SSH Interface. This configuration is marketed and sold as the GSW SSH Server

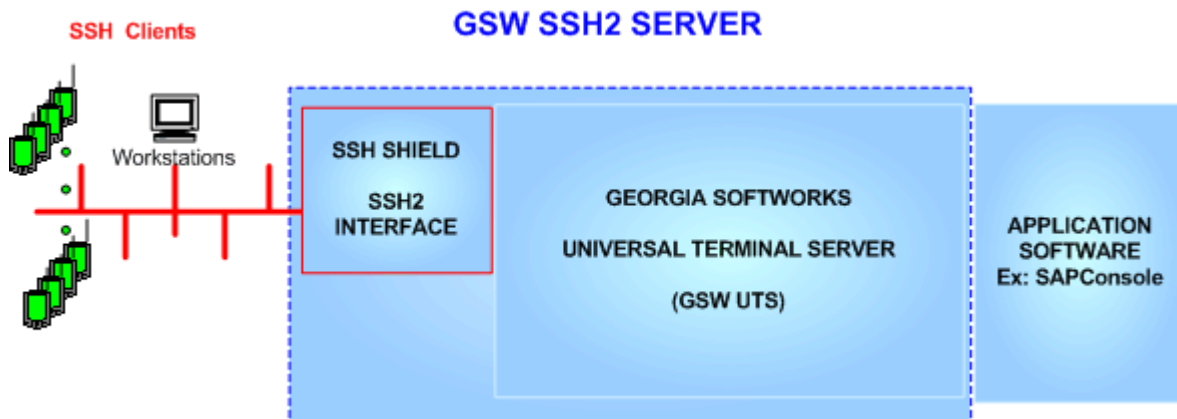


Figure 3: GSW SSH Server Block Diagram

## Installation

### Overview

When you purchased the GSW SSH Server you either:

- a. Own a GSW Telnet Server (UTS) and are upgrading to the SSH Server
- OR**
- b. A new customer purchasing the GSW SSH Server<sup>4</sup>.

If you own a GSW Telnet Server and are upgrading to the SSH Server then:

- a. You must have GSW Telnet Server Version 6.50 or higher to install the SSH Shield. The Telnet Interface becomes disabled when the SSH Shield is installed. If you have an older version then you will need to upgrade to the Version 6.50 or higher before you can apply the SSH Shield.
- b. Next install the GSW SSH Shield
- c. Register the GSW SSH Server.

If you are purchasing a new GSW SSH Server then:

- a. You will receive the current version of the GSW Telnet Server. Install the GSW Telnet Server according to the Installation Instruction in the GSW UTS User Manual. You do not need to register the Telnet Server at this time. Registration takes place after the installation of the GSW SSH Shield.
- b. Next install the GSW SSH Shield
- c. Register the GSW SSH Server.

**NOTE:** The GSW SSH Server requires registration. The registration for the GSW UTS is not sufficient for the GSW SSH Server.

---

<sup>4</sup> In conjunction with the GSW UTS Server

## Procedure

Installation of the GSW SSH Server software is simple and quick. From Windows 7/8/2008/R2/2012/R2/2016/2019, NT/XP/VISTA/2000/2003, perform the following:

1. Run the setup program (sshshld.exe). The Welcome screen of the setup program is displayed and you are reminded and urged to exit all windows programs before continuing. You are also reminded that you **must have administrative privileges** to install this program. Click **Next**.



Figure 4: Installation Welcome Screen

1. A screen is displayed indicating the folder where the GSW SSH Shield will be installed. The default is:

C:\Program Files\Georgia SoftWorks\Georgia SoftWorks SSH SHIELD.

You may change the installation directory at this time. *Note: Make sure that the users of the SSH Server have full access to the installation directory.*



Figure 5: Installation – Choose Destination Folder

Select the Program Folder for the SSH Server. **Click Next.**

2. A shell opens a window with installation status lines similar to the figure below.

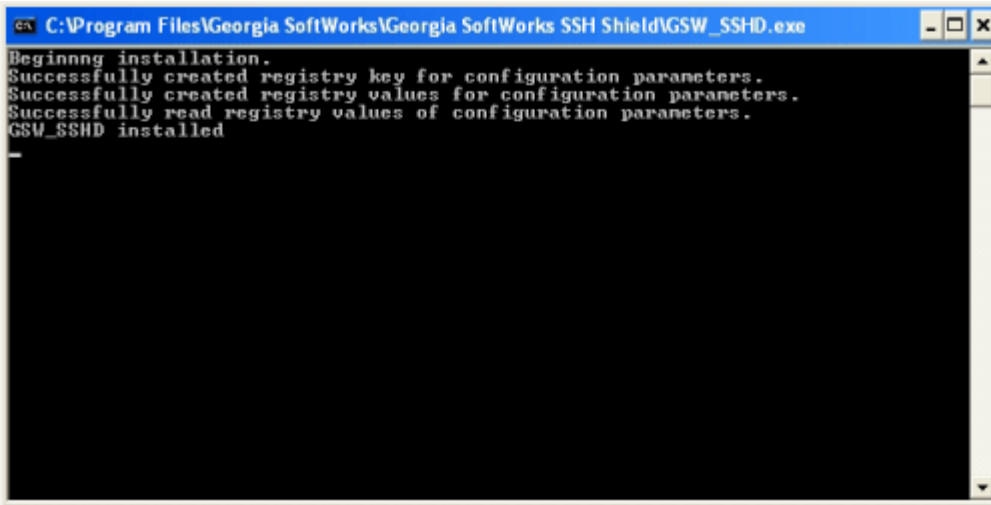


Figure 6: Installation – Command Shell Status Lines

3. Now the Setup is complete! Click Finish and *Now it's time to register the SSH Server!*

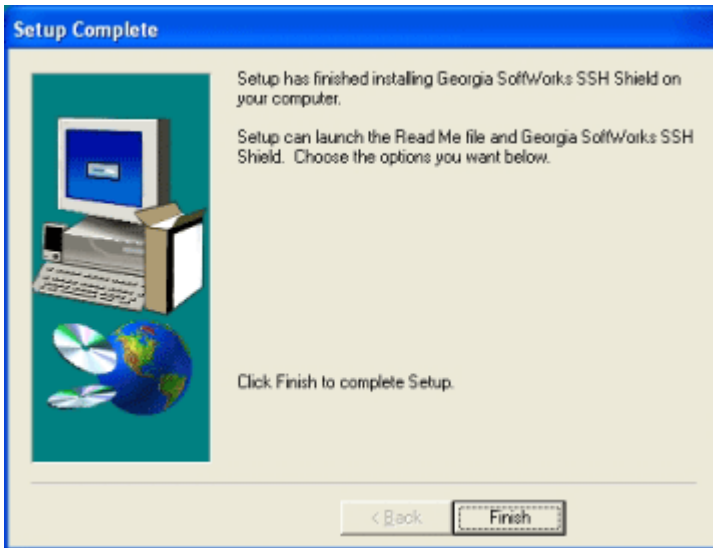


Figure 7: Installation Complete

Please view the `readme.txt` file as it may contain late breaking information about the SSH Server that has not yet made it into the User Manual. Release notes are also contained in the `readme.txt` file.

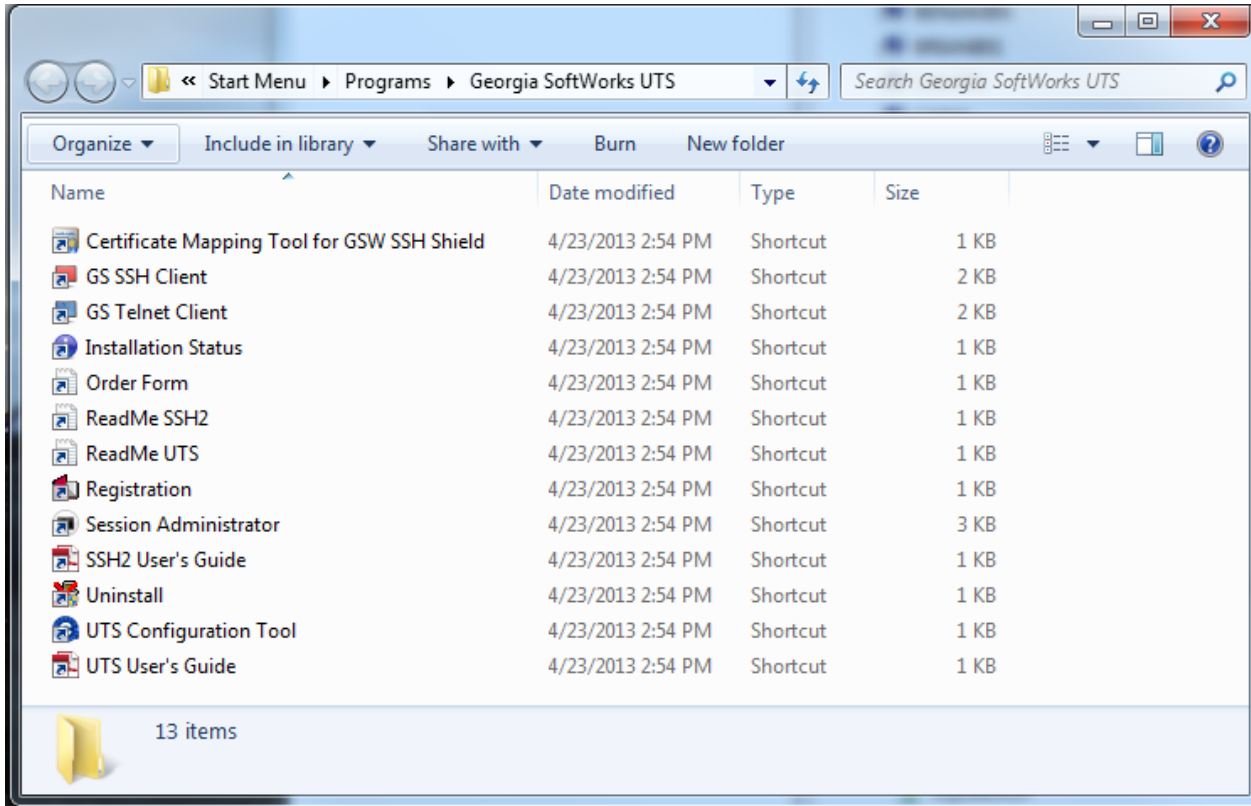


Figure 8: GSW UTS Program Group

Installation will result in the Georgia SoftWorks program group item “Installation Status” showing GSW SSH as installed. Additionally, the version of the GSW SSH Shield is displayed along with the status of the server and other Georgia SoftWorks software that may be installed.

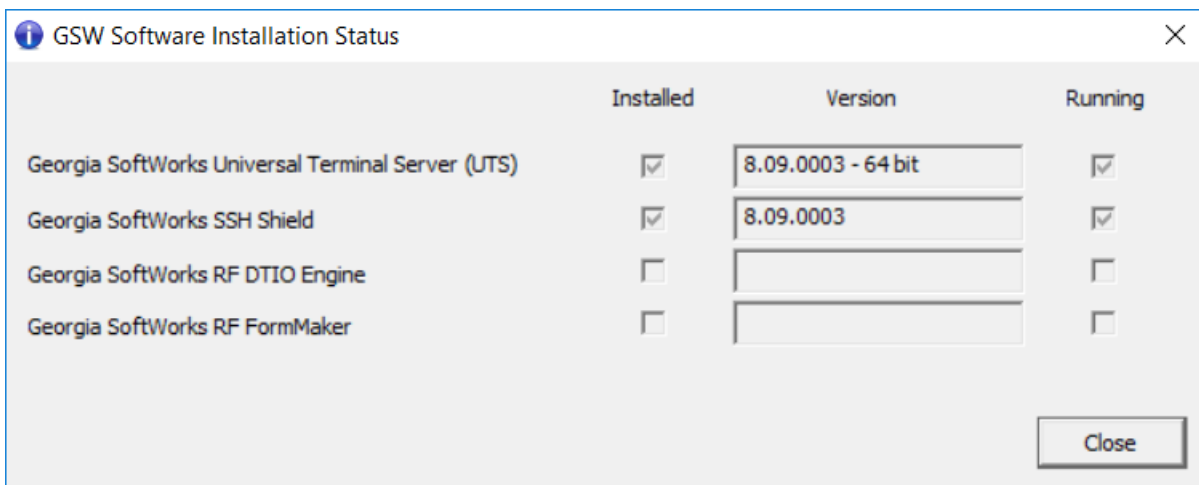


Figure 9: SSH Installation Status (Your version will be shown in the fields above)



## Registration

The GSW SSH Server is licensed for a single server. The license must be *activated* for the software to operate. To activate the license a valid *Serial Number* is required and is examined periodically by the SSH Server software. The Serial Number also allows new versions to be downloaded and installed for the duration of your subscription plan.

Two methods exist to obtain a valid Serial Number.

1. Registration via Software Serial Number.

This method exists for environments that do not support Parallel or USB ports. In brief this entails providing GSW with a machine specific Product ID. A Serial Number is generated based on the Product ID. This is usually performed via the GSW Ticket System, however in some cases email, fax or telephone. See page 24 for details on Software registration.

2. Registration via Floating License

The Serial Number is pre-programmed into a specific hardware key that came with your purchase. The hardware key connects to a parallel or USB port on the server. See page 24 for details on registration via the Floating License.

### Registration via Software Serial Number

To run the GSW SSH Server you must first register the software. (*This registration is NOT required if you installed the Floating License, Page 24*) Registration via Software Serial Number entails just a few steps that involve obtaining the Product ID and providing this Identification to Georgia SoftWorks so a Serial Number can be generated. Georgia SoftWorks will provide you with the Serial Number based on the Product ID. When you enter the Serial Number into the Registration Tool, click Register.

**NOTE:** Read System Signature chapter at the end of manual (page 115).

#### How to Register the Software

To run the registration software -

- Select the *Start* button on the task bar; select *Programs*, then *Georgia SoftWorks UTS Server* and right click on *Registration* and Run as Administrator.

Prior to registering the SSH Server, a reminder dialog is presented indicating that the SSH Shield is not registered.

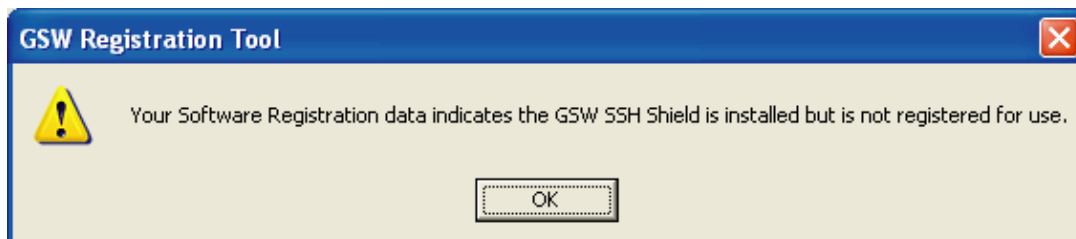


Figure 10: Registration – SSH Shield is not registered for use

The GSW SSH Server will be fully functional for a Trial Period of 30 days without requiring registering when installed for the first time on a system. *Click OK*

**IMPORTANT NOTE:** If you already own a GSW Telnet Server and you want to run a 30 day trial of the GSW SSH Server then you will need to request a 30 day trial serial number from Georgia SoftWorks. Please save a copy of the current SERIAL NUMBER for your telnet server prior to installing a 30 day trial GSW SSH Server. In the event that you do not purchase the GSW SSH Server prior to the expiration of the trial, you will need to apply your original serial number to re-activate the original GSW Telnet Server.

Next, the registration screen is displayed. The Registration program automatically fills in the Product Information fields as shown in the figure below. Complete the Customer Information fields as shown in the figure below.

**Note:** The Product Information *Name* and *Version* must contain valid data or it will not generate a correct Product ID.

GSW Registration Tool Ver. 1.27.00.0010 - LogisticsNY

Customer information

Name: Captian Secure

Company: ACME Battleships

Street Address 1: ATLANTIC OCEANS

Street Address 2:

City: Port Secure

State: GA Zip: 30534

Country: USA

Phone: 706.265.1018

Fax: 706.265.1020

Purchased From: Georgia SoftWorks

Application software: Titanium Security

Product information

Name: GSW\_UTS Sessions Requested: 3000

Version: 8.09 Zone: 8aYWx28p

Product ID:

3CF4AF6F7310DCA047770023D223AE0057D4C2171346

Registration information

Please enter your serial number in the window below and click on the 'Register' button

Expiration date: Not set

Free updates until: Not set

Parameter:

Register

Save to file... Print... Hw Key... Close

Figure 11: GSW Registration - Initial Screen

Note that the Customer Information and Serial Number in the Registration Information may be already filled. This will be the case if the GSW UTS has previously been registered and operating as the GSW Telnet Server.

The registration information must be provided to Georgia SoftWorks to obtain the Serial Number. Several methods are available for your convenience.

1. Please complete the *Customer Information*, including the *Purchased From* and the *Application software* fields in the Registration Screen.

The registration information must be provided to Georgia SoftWorks to obtain the Serial Number. Several methods are available for your convenience.

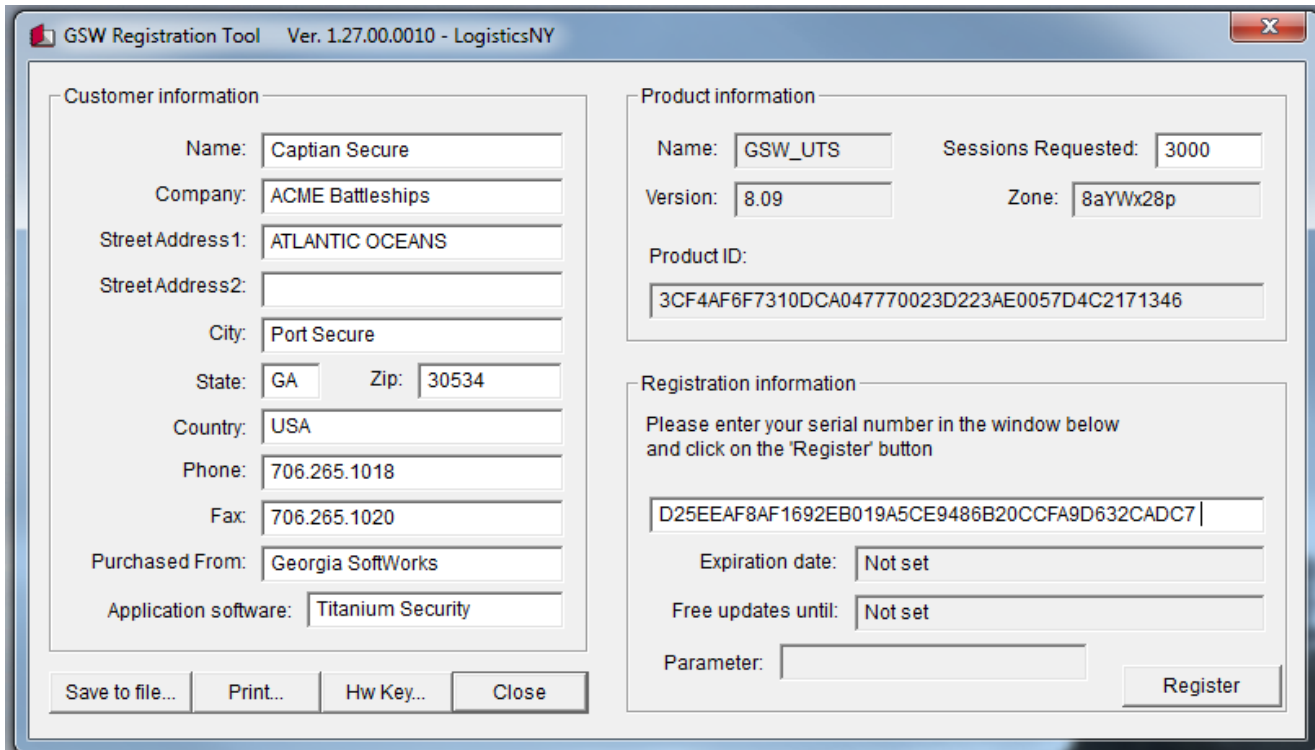
- Go to: [http://www.georgiasoftworks.com/support\\_ost/open.php](http://www.georgiasoftworks.com/support_ost/open.php) to submit a ticket for Registration. Complete necessary fields and attach the file you saved in the previous step. – **Fastest response and preferred method.**

## OR

- Email the file to [registration@georgiasoftworks.com](mailto:registration@georgiasoftworks.com)

Once Georgia SoftWorks receives the information, we can generate a Serial Number on demand and will send it to you. You may close the registration program at this time.

- When the Serial Number is provided run the Registration Program (see page 24) again and enter the Serial Number. The easiest method to get the serial number is to highlight the returned Serial Number and copy (ctrl-c). Then position the mouse in the Serial Number field in the Registration Information box and paste (ctrl-v).



The screenshot shows the 'GSW Registration Tool' window with the following data:

Customer information		Product information	
Name:	Captian Secure	Name:	GSW_UTS
Company:	ACME Battleships	Sessions Requested:	3000
Street Address 1:	ATLANTIC OCEANS	Version:	8.09
Street Address 2:		Zone:	8aYWx28p
City:	Port Secure	Product ID:	3CF4AF6F7310DCA047770023D223AE0057D4C2171346
State:	GA	Zip:	30534
Country:	USA		
Phone:	706.265.1018		
Fax:	706.265.1020		
Purchased From:	Georgia SoftWorks		
Application software:	Titanium Security		

**Registration information**

Please enter your serial number in the window below and click on the 'Register' button

D25EEAF8AF1692EB019A5CE9486B20CCFA9D632CADC7 |

Expiration date: Not set

Free updates until: Not set

Parameter: [ ]

Buttons: Save to file..., Print..., Hw Key..., Close, Register

Figure 12: Registration - Serial Number Applied

4. Click Register.

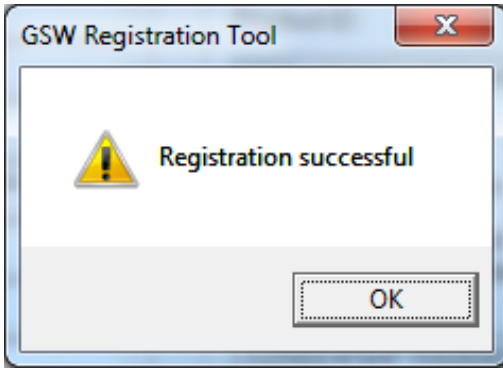


Figure 13: Registration Successful Screen

5. Click OK.

Now the software is registered.

You will notice that in this case the Parameter field in the registration form is set to 3000, SSH Shield. This indicates that the SSH Server is installed and registered and is enabled for 3000 sessions.

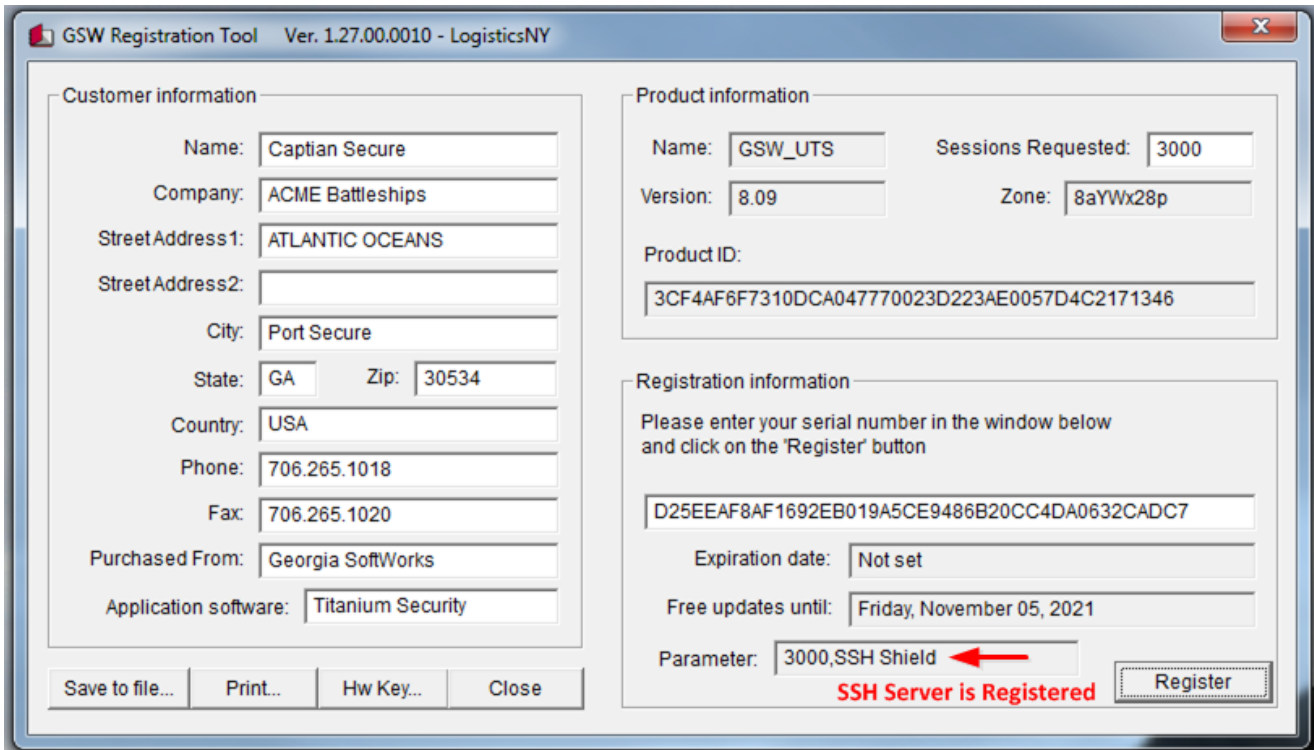


Figure 14: Registration Verification

If you have purchased the Federal Information Processing Standards Publications (FIPS 140-2) option, you can verify that it is enabled by viewing the registration screen as shown below in Figure 15. Please note that the GSW SSH Server must be installed for the FIPS option to be available. GSW True FIPS 140-2 compliant connections can be identified using the GSW Session Administrator in the GSW UTS Server. Please see the GSW UTS Users Guide for further details.

The screenshot shows the 'GSW Registration Tool' window with the following fields:

- Customer information:** Name: Captian Secure, Company: ACME Battleships, StreetAddress1: ATLANTIC OCEANS, City: Port Secure, State: GA, Zip: 30534, Country: USA, Phone: 706.265.1018, Fax: 706.265.1020, Purchased From: Georgia SoftWorks, Application software: Titanium Security.
- Product information:** Name: GSW\_UTS, Sessions Requested: 3000, Version: 8.09, Zone: 8aYWx28p, Product ID: 3CF4AF6F7310DCA047770023D223AE0057D4C2171346.
- Registration information:** Please enter your serial number in the window below and click on the 'Register' button. Serial number: D25EEAF8AF1692EB019A5CE9486B20CC4DA0632C8DB5. Expiration date: Not set. Free updates until: Friday, November 05, 2021. Parameter: 3000,SSH Shield,FIPS. A red arrow points to the parameter field, and the text 'FIPS 140-2 is Enabled' is displayed in red below it.

Figure 15: Registration - Verify that FIPS 140-2 is Enabled

**IMPORTANT:** READ SYSTEM SIGNATURE CHAPTER AT END OF MANUAL (page 115).

You may now run the Georgia SoftWorks SSH Server. Note that you will be able to obtain Free Updates until the date specified.

### Floating License – Overview

The Georgia SoftWorks Floating License provides the flexibility to rapidly move the GSW SSH Server from one machine to another. *If you are unable to use the Floating License - skip this section and go to the section on Registration via Software Serial Number on page 24.*

**NOTE:** When a SSH Server Pack is purchased (SSH Server and GSW Telnet Server), the same physical Floating License will contain valid Serial Numbers for both products.

With the Floating License **NO** software registration is required for the SSH Server to operate.

Common scenarios where the Floating License is useful include:

- **Laboratory usage in a development or test environment** where the SSH Server is required for short periods of time on any particular machine and then moved to a new machine.
- **Backup Servers in a production environment.** Typically multiple SSH Servers are purchased for backup systems, however with a Floating License the Hardware Key can be quickly moved from the primary machine to the backup without any other registration requirements.
- **Environments where a failed server must be replaced or rebuilt and immediately restored to operation with full SSH Server capability.**

The Georgia SoftWorks Floating License is a hardware key that connects to a *female* parallel port connector or USB Port on the server. The parallel port Floating License does not impact functionality of the port for other uses. The parallel hardware key acts as a pass-through allowing normal connections to the other side of the key.

The Georgia SoftWorks Floating License is a hardware key that can be ordered for a Parallel or USB Port.




Parallel Port Floating License	USB Floating License
 <p>Figure 16: Floating License – Parallel Port</p> <p>The Parallel Port Floating License is a Pass Through allowing normal function of the port.</p>	 <p>Figure 17: Floating License - USB Port</p> <p>Not attached to a Server</p>
<p>The Parallel Port Floating License connects to a <b>female</b> parallel port on the server and does not impact functionality of the port for other uses. It acts as a pass though allowing normal connections to the other side of the key.</p>	 <p>USB LED Lights when Installed</p>

Figure 18: Floating License - Hardware Key

The SSH Server will recognize the presence of the key and activate the software with the proper date for which free version upgrades can be obtained. It does not matter which parallel or USB port on the server the Hardware Key is installed, as all ports will be scanned for the installation of the key.

The Floating License currently is installed using the manufacturer SafeNet, previously Aladdin of the hardware key setup program. It is described below. The name of the hardware key is HASP4 and you will see it displayed in the setup screens. The best drivers for the HASP4 are the HASP HL drivers.

## Floating License – Hardware Key Installation Instructions

**Note:** If you are using a *USB Floating License on a Windows NT system* - run the file aksnt4usb.exe prior to the following steps.

1. Copy the files from the Floating License folder (hardkey) to the hard drive on your server.
2. Run the HASPUserSetup.exe program and follow the installation instructions. After installation of the hardware key install the GSW SSH Server as described on page **Error! Bookmark not defined.** (if it is not already installed). See the GSW SSH Server User's Guide for installation instructions of the GSW SSH Server.
3. If you have User Account Control enabled you may get a prompt that says "Do you want to allow the following program to make changes to this computer?" Click Yes

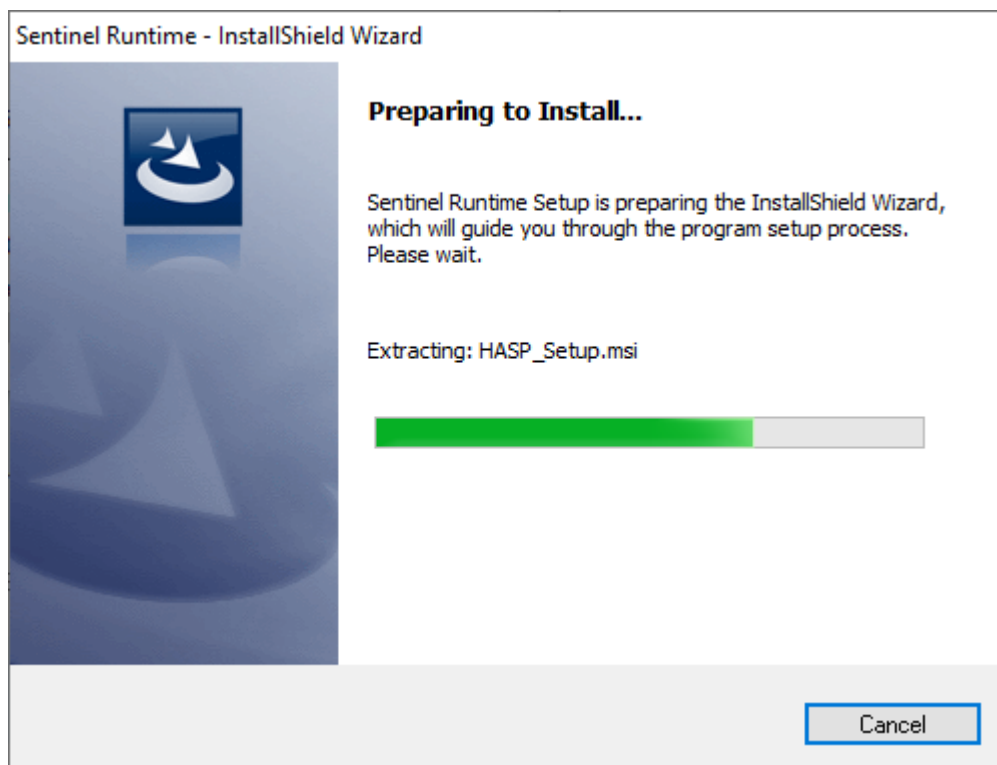


Figure 19: Hasp Preparing to Install



- 4. You will first see the gemalto (formerly SafeNet) initial Welcome Screen.



Figure 20: Sentinel welcome screen

### Click Next

- 5. The next screen displayed is the gemalto License Agreement screen.

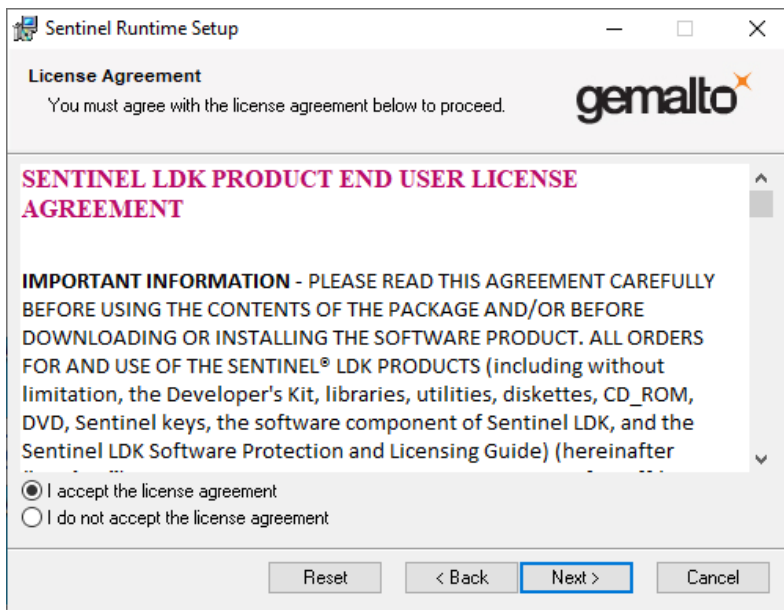


Figure 21: SafeNet License Agreement

Read the license agreement and select “I accept the license agreement”

### Click Next

6. Ready to Install Sentinel Runtime Setup

**Click Next.**

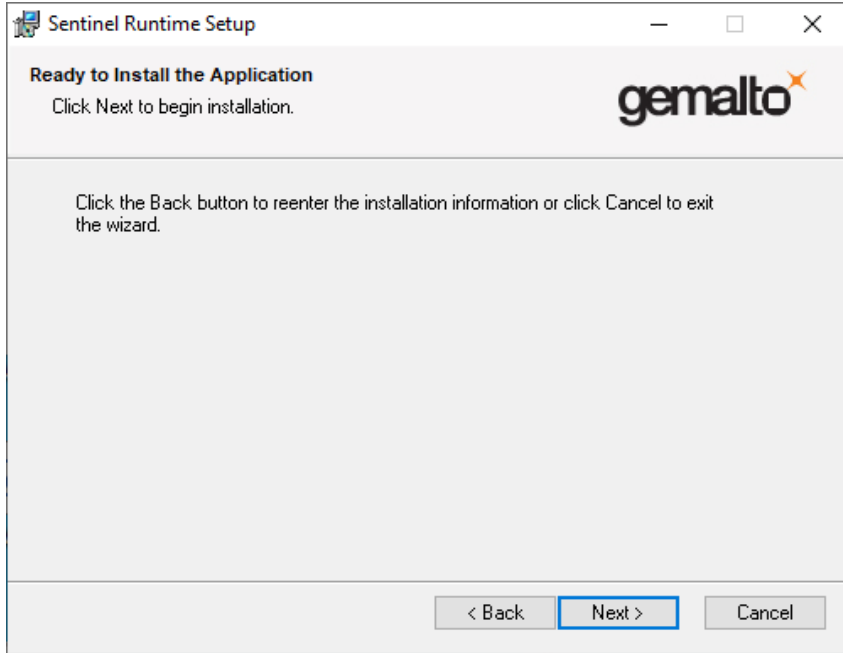


Figure 22: gemalto Sentinel Runtime Setup

7. Install Drivers - Progress bar, updating system.

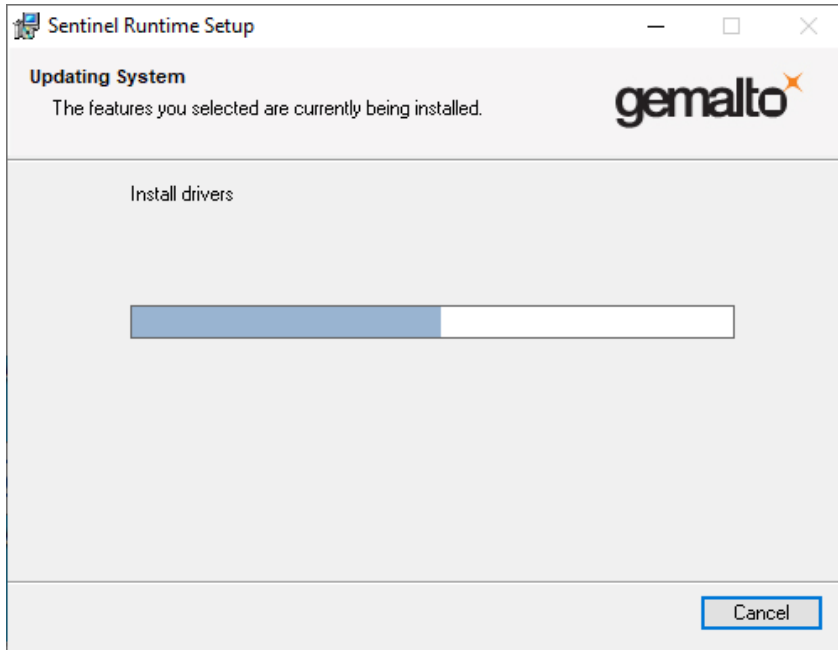


Figure 23: gemaltor Sentinel Runtime Setup Progress bar

## 8. Gemalto Sentinel Successfully Installed.

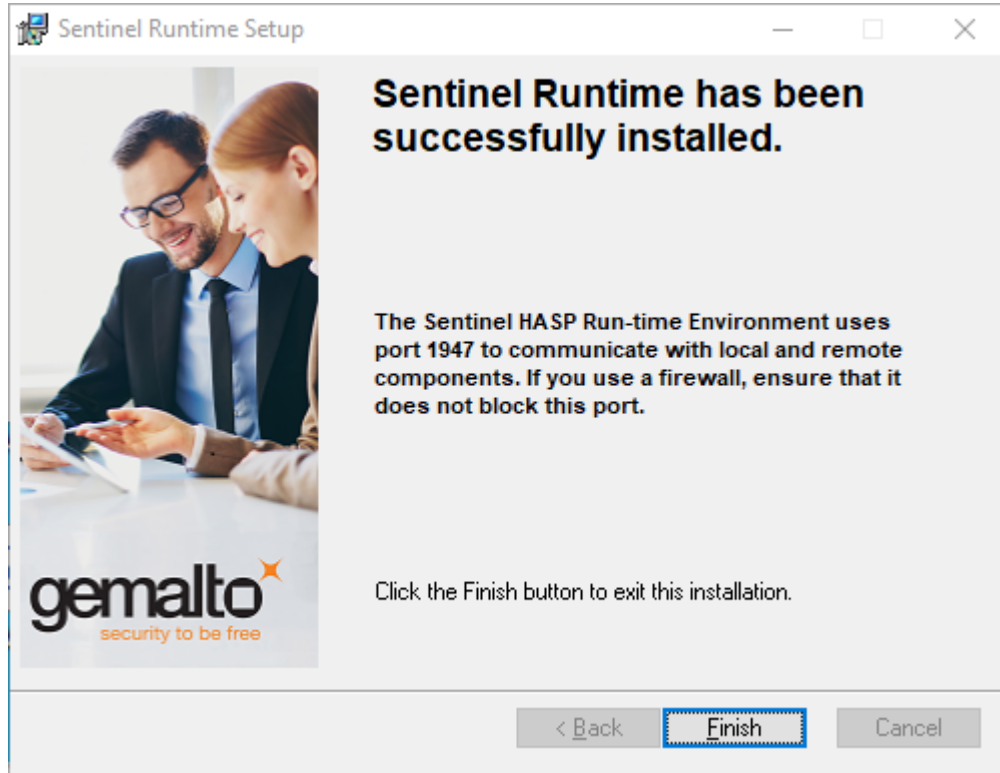


Figure 24: SafeNet Validating Install

**Click Finish.**

## 9. Plug the hardware key onto the parallel or USB port on the server.

NOTE: On some systems you may have to reboot the server after installation. If the Floating License is not recognized (by the UTS) after installing the driver, please reboot the server.

### **Uninstall Floating License – (Hardware Key)**

In the event that you need to uninstall the Floating License (SafeNet HaspHL) please use the Windows Control Panel Add/Remove Programs administrative utilities.

**NOTE: Removing or uninstalling the Floating License will disable the GSW UTS Server.**

## GSW SSH Server

After Installation and Registration the GSW SSH Server is ready to use.

You can further configure the SSH Server to use more advanced features as needed. See page 42. Power configuration options for the SSH Server are implemented as common Universal Terminal Server configuration parameters. See User Manual for the [GSW Universal Terminal Server](#) for information on the powerful features available to the GSW SSH Server.

Using the Installation Status Program Item within Georgia SoftWorks UTS program group, you can view the Installation Status of the GSW UTS and SSH Server.

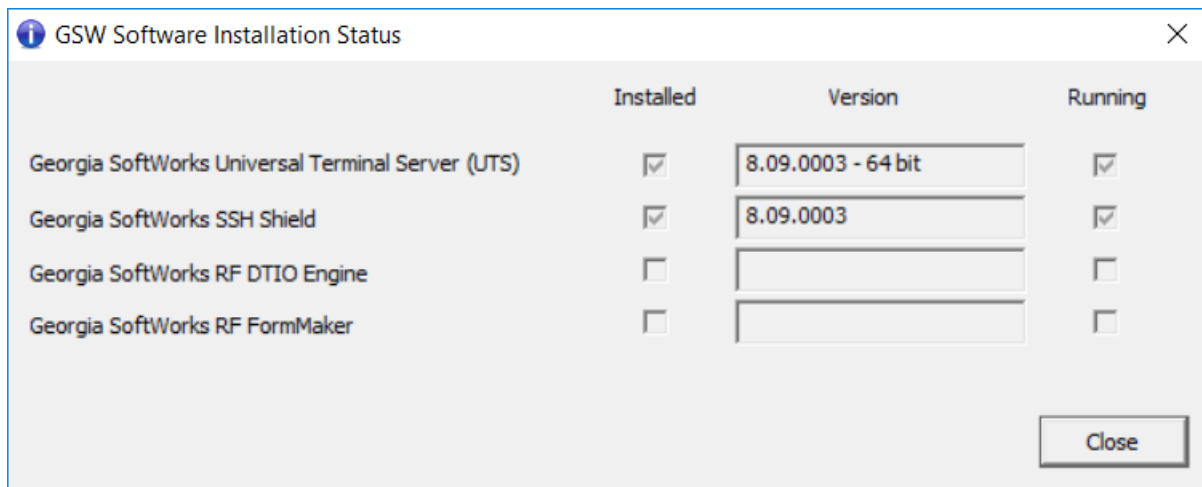


Figure 25 - GSW Software Installation Status Tool

The Windows Services Utility can be used to view and alter the status of the GSW SSH and the GSW UTS services.

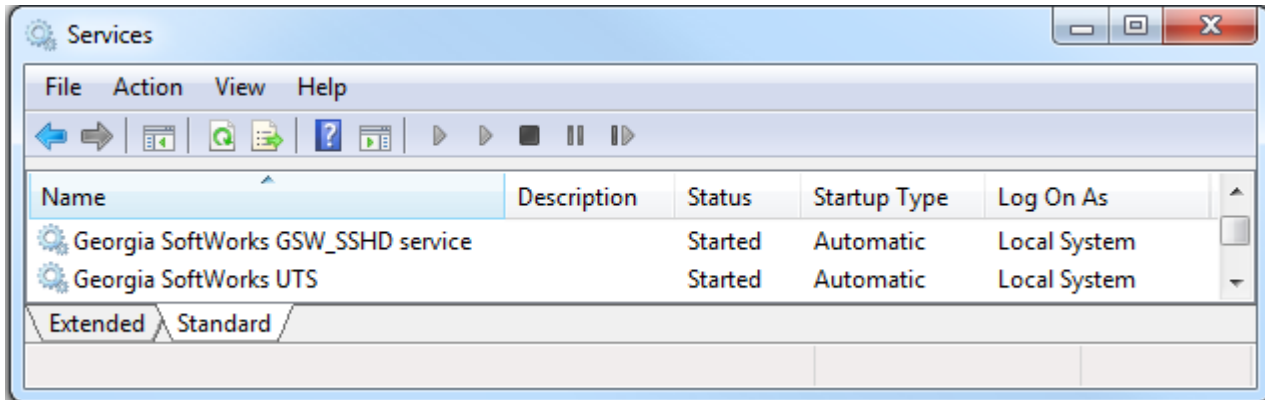


Figure 26: Control Panel - GSW SSH Services Started

The Georgia SoftWorks GSW\_SSHD service and the Georgia SoftWorks Universal Terminal Server should both have a status of Started and a Startup Type of Automatic.

Using the Windows Services utility is the recommended method to start and stop the GSW services when required.

## GSW FIPS 140-2 Compliant Option

GSW provides a Federal Information Processing Standards Publication (FIPS) 140-2 compliant option for those entities with requirements to meet cryptographic module security standards to protect sensitive and valuable data. FIPS standards are either mandated or recommended for use in federal government information technology (IT) systems.

Georgia SoftWorks undertook a purposed and specific development effort in order to provide required FIPS 140-2 compliant SSH server and client software to the United States Military. Having completed this task, GSW is able to make this software available to other branches of the Federal government as well as State governments and other institutions including research, educational and commercial.

### Software Requirements

In addition to the development required for FIPS 140-2 compliance of the GSW server and client software, the GSW mobile clients must run on an operating system that is FIPS 140-2 certified or provides a cryptographic module that has been certified.

In order for your SSH connections to be FIPS 140-2 compliant you must ensure that you have the minimum GSW software versions as well as the proper Windows Mobile/CE operating system version.

#### Software Requirements for FIPS Compliancy

GSW Software	Version		Certificate
GSW UTS Server	7.50+		<a href="#">#918</a>
GSW SSH Server	7.50+		<a href="#">#918</a>
GSW Desktop Clients	7.50+		<a href="#">#918</a>
GSW CE/Mobile Clients	7.50+		<a href="#">#560</a> , <a href="#"># 825</a>

Table 1: GSW Software versions required for FIPS 140-2

Required Device Operating System for Mobile/CE Clients			Certificate
Windows CE 5.0 Depends on Vendor <i>- Made available to OEMs via Windows Update 061211_KB911762</i>			<a href="#">#560</a>
Windows Mobile 5.0			<a href="#">#560</a>
Windows CE 6.0			<a href="#">#825</a>
Windows Mobile 6.0+			<a href="#">#825</a>

Table 2: Device Operating System Versions Required for FIPS 140-2

The significant aspect of the client device operating system is that the version of the cryptographic module rsaenh.dll must be NIST (National Institute of Standards and Technology) certified, which begins with build 14343.0.0. With Windows CE 5.0 extra attention should be taken to ensure the version of rsaenh.dll. This may require contacting the device vendor to determine the correct version number of that cryptographic module.

### Enable Option

FIPS 140-2 must be enabled on both the GSW SSH server and the GSW clients to complete a FIPS 140-2 compliant connection.

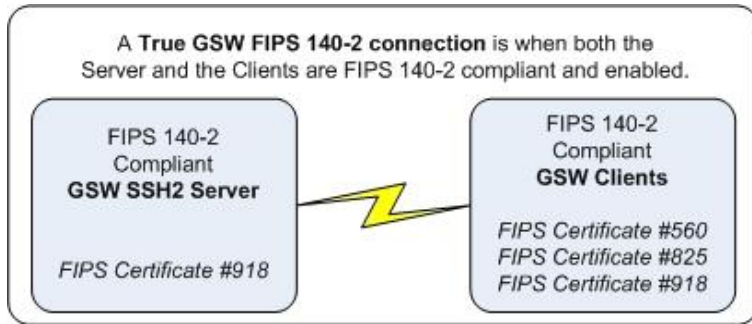


Figure 27: GSW True FIPS 140-2 Connection – Server and Client

### ENABLE FIPS 140-2 ON SSH SERVER

Proper registration will enable the FIPS option on the SSH Server. View the registration tool to ensure the GSW SSH Server is registered with the FIPS option enabled. Select the Start button on the task bar; select Programs, then Georgia SoftWorks UTS and then Registration. The current registration information is displayed.

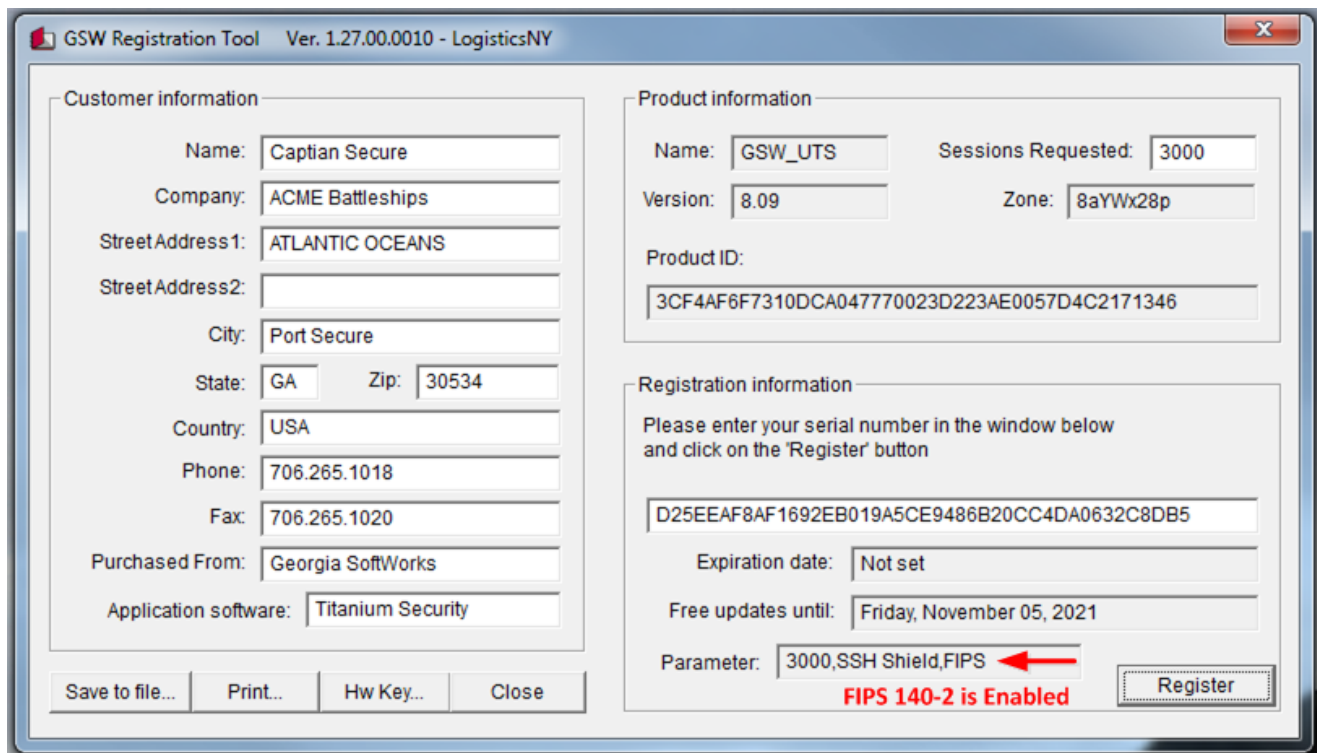


Figure 28: FIPS 140-2 Option Enabled

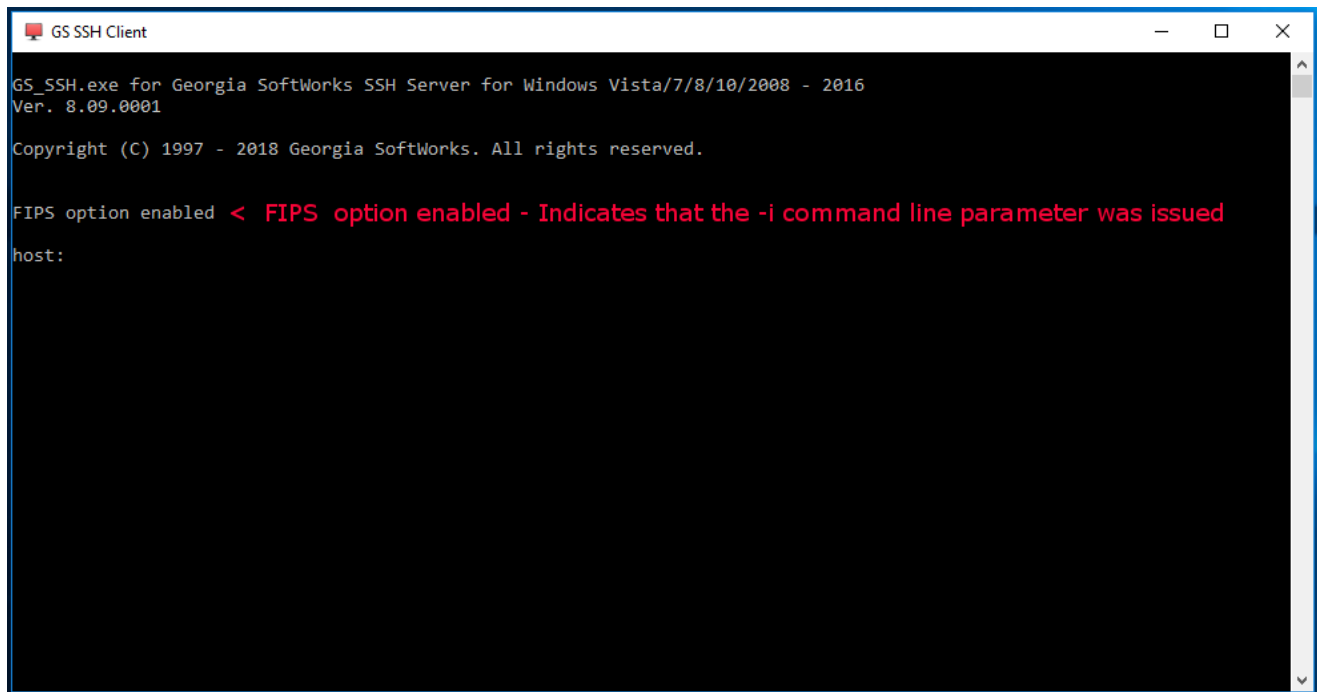
In the Parameter field you will observe the number of concurrent sessions allowed followed by the text “SSH Shield” indicating that the GSW SSH server is licensed and FIPS indicating that the FIPS 140-2 option is enabled.

## ENABLE FIPS 140-2 ON GSW MOBILE/CE and DESKTOP CLIENTS

### GSW Desktop Client

Use the "-i" command line parameter when launching on GSW Desktop clients to enable FIPS 140-2 option. Please see the [UTS user's guide](#) for a description and examples of desktop client command line options.

When FIPS 140-2 enabled GSW desktop clients are launched you will receive a banner indicating that the "-i" command line parameter was issued by the client.

A screenshot of a terminal window titled "GSW SSH Client". The terminal output shows the following text:

```
GS_SSH.exe for Georgia SoftWorks SSH Server for Windows Vista/7/8/10/2008 - 2016
Ver. 8.09.0001
Copyright (C) 1997 - 2018 Georgia SoftWorks. All rights reserved.

FIPS option enabled < FIPS option enabled - Indicates that the -i command line parameter was issued
host:
```

Figure 29: Desktop Client "-i" option issued

Please note that to have a both ends (client and server) FIPS 140-2 compliant, FIPS 140-2 must be enabled on the GSW SSH Server too.



### GSW Windows Mobile/CE Clients

Enable FIPS140-2 on GSW Mobile/CE clients via the Encryption list box. The Mobile/CE device screen that you see will be similar to the ones below.

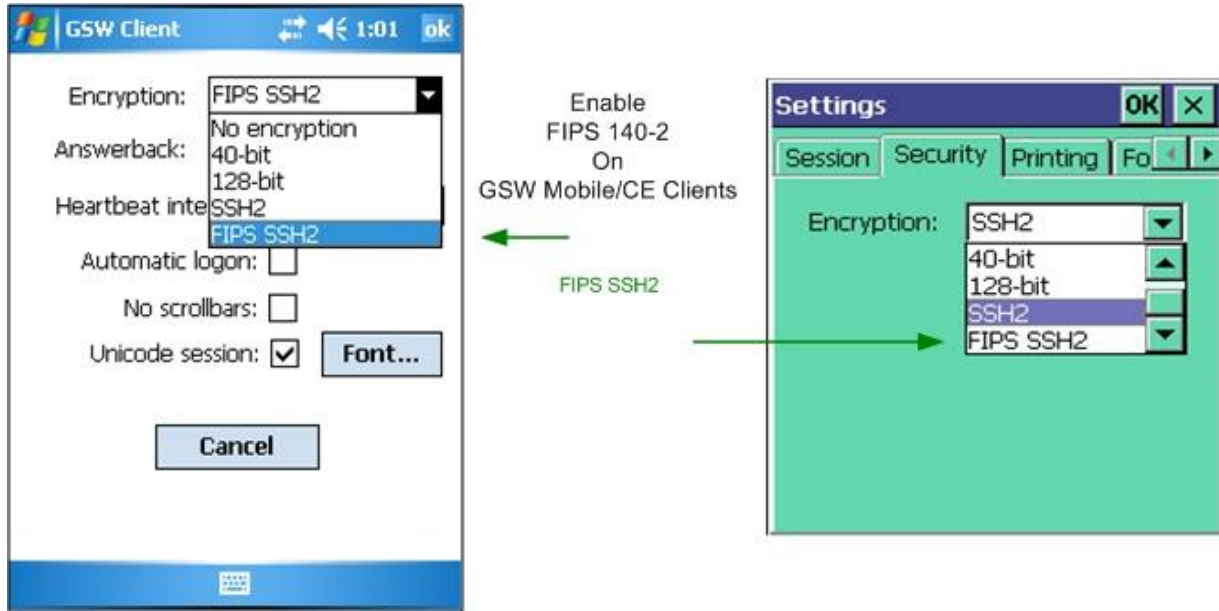


Figure 30: Enable FIPS 140-2 on GSW Mobile Clients

Please note that to have a both ends (client and server) FIPS 140-2 compliant, FIPS 140-2 must be enabled on the GSW SSH Server too.

## FIPS 140-2 Connections

Using the UTS Session Administrator you can verify True GSW FIPS 140-2 compliant connections. An asterisk “\*” will be prepended to the user name for connections that are FIPS 140-2 compliant for both the client and the server.

The possibility exists that a third party client may be FIPS 140-2 compliant but it cannot be verified unless it is a GSW client.

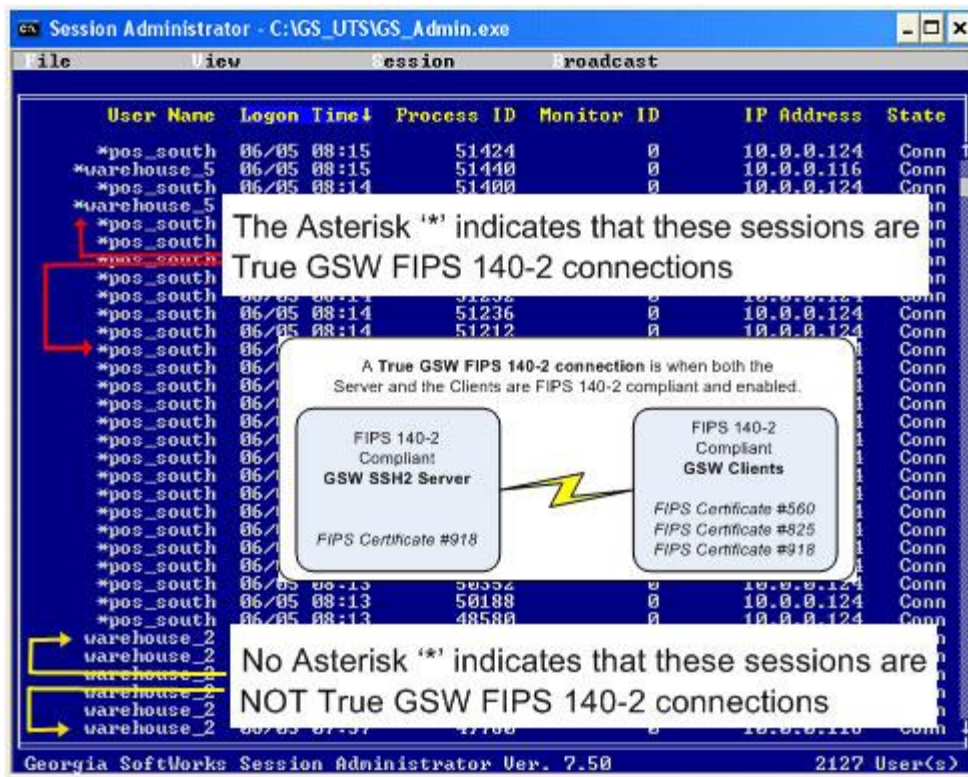


Figure 31: Verify FIPS 140-2 Compliant Connections

## Configuration

No configuration is required beyond installation in order for the GSW SSH Server to operate providing secure logon, strong encryption and data integrity on an insecure network. Optional SSH Configuration is provided to implement advanced features. The GSW SSH Server reads configuration values each time the GSW\_SSHD service is started.

Please consider the optional [GSW UTS GUI Configuration tool](#) for SSH provisioning or use the methods described below.

### Registry Key Locations

Registry keys referenced in this User's Guide are located here on 64 bit operating systems

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters
```

Registry keys referenced in this User's Guide are located here on 32 bit operating systems

```
HKEY_LOCAL_MACHINE\SOFTWARE\Georgia SoftWorks\GSW_SSHD\Parameters
```

### Allow user to use SCP channel

Allow/Disallow users using transfer files using the Secure Copy protocol. The use of the "scp channel" is defined as defined in the SSH Connection Protocol.

Example using PuTTY's secure copy is:

```
pscp -l mike -pw (zPro2@@5) -2 -v tt.txt mike@gsw2003:pp.txt
```

You can Allow/Disallow this capability by modifying the following registry key.

This configuration is contained in the registry key bAllowServiceSCP. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\GeorgiaSoftWorks\GSW_SSHD\Parameters\bAllowServiceSCP
```

The default value is 0x0 (Do not allow Secure Copy protocol). The value 1 allows the use of the Secure Copy Protocol.

Note that szSFTPRoot (page 44) must be set for SCP to work.

The following is the procedure to change the registry key for allowing the use of the "scp channel."

1. Click the **Start** button at the bottom left corner of your screen.

2. Click **RUN**

3. Type REGEDIT

4. Click **OK**

5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\GeorgiaSoftWorks\GSW_SSHD\Parameters\bAllowServiceSCP
```

6. Select the menu item **Edit** and then click on **Modify**

7. Enter the new value for the bAllowServiceSCP and click **OK**

*The new value will take effect when the GSW SSHD service is restarted.*

## **Allow user to start a remote Shell**

Allow/Disallow users use of the ‘shell’ channel as defined in the SSH Connection Protocol. This functionality may be accessed using ssh client software.

You can Allow/Disallow this capability by modifying the following registry key.

This configuration is contained in the registry key bAllowServiceShell . The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowServiceShell
```

The default value is 0x01. (Allow remote shell). The value 0x00 disallows the use of a remote shell.

The following is the procedure to change the registry key for allowing the use of a “shell.”

1. Click the **Start** button at the bottom left corner of your screen.

2. Click **RUN**

3. Type REGEDIT

4. Click **OK**

5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\GeorgiaSoftWorks\GSW_SSHD\Parameters\  
bAllowServiceShell
```

6. Select the menu item **Edit** and then click on **Modify**

7. Enter the new value for the bAllowServiceShell and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Allow user to use SFTP subsystem

Allow/Disallow users use of the SFTP subsystem as defined in the SSH Connection Protocol. This functionality may be accessed using sftp client software.

You can Allow/Disallow this capability by modifying the following registry key.

This configuration is contained in the registry key `bAllowSFTP`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowSFTP
```

The default value is `0x00`. (Disallow use of SFTP subsystem). The value `0x01` allows the use of a sftp subsystem.

The following is the procedure to change the registry key for allowing the use of the sftp subsystem.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowSFTP
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `bAllowSFTP` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Specify the SFTP and SCP Root Folder

Specify the root folder for SFTP and SCP (page 42). This parameter must point to a valid local folder. For example `C:\sftp\root`.

The user's sftp path is set to this value with the domain name and user name appended.

Domain: receiving

User name: bob

For example `C:\sftp\root\receiving\bob`.

Note: The system administrator must ensure that `receiving\bob` has access to this folder and its subfolders.

This configuration is contained in the registry key `szSFTPRoot`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szSFTPRoot
```

The default value is `NOT SET`.

The following is the procedure to change the registry key for specifying the SFTP Root Folder..

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type `REGEDIT`
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szSFTPRoot
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szSFTPRoot` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Specify UNC path for SFTP and SCP

Specify Universal Naming Convention (UNC) destinations for SFTP and SCP (page 42). UNC paths start with double slashes or backslashes and you can copy files with the security of SSH to network shares.

There are three registry values that must be configured on the GSW SSH Server to use a UNC destination for SFTP or SCP.

- bAllowServiceSCP
- bAllowServiceSFTP
- bRestrictedSFTP
- szSFTPRoot

The registry keys will be set to:

- Set bAllowServiceSCP to 1 (Enables SCP operation)
- Set bAllowSFTP to 1 (Enables SFTP operation)
- Set bRestrictedSFTP to 0 (Disables username and domain restrictions)
- Set szSFTPRoot to \\ (Sets destination to use a UNC path)

All three of the registry values are located here:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters
```

The following procedure is used to change the registry keys for specifying a UNC destination.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szSFTPRoot
```

Select the menu item **Edit** and then click on **Modify**

Enter \\ (just the backslashes) for szSFTPRoot and click **OK**

6. Select Registry Key:

---

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bRestrictedSFTP
```

Select the menu item **Edit** and then click on **Modify**

Enter **0** for the bRestrictedSFTP and click **OK**

7. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowServiceSCP
```

Select the menu item **Edit** and then click on **Modify**

Enter **1** for bAllowServiceSCP and click **OK**

8. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowSFTP
```

Select the menu item **Edit** and then click on **Modify**

Enter **1** for bAllowSFTP and click **OK**

*The new values will take effect when the GSW SSHD service is restarted*

Network shares should be created for the destinations for sftp/scp.

For example create a share \\backups\USA\Tampa with the name TAMPA and another share \\backups\USA\Atlanta with the name Atlanta

Some examples using PuTTYs pscp command the results status.

```
pscp -scp C:\GS_UTS\GS_Auto.txt johnny@jude:192.168.1.56\ATLANTA
GS_Auto.txt | 1 kB | 1.7 kB/s | ETA: 00:00:00 | 100%
```

```
pscp -sftp -v C:\GS_UTS\^*.txt xfer@srv-2016:srv-2016\Tampa
ColorMap.txt |14 kB | 14.3 kB/s | ETA: 00:00:00 | 100%
gsnull.txt | 0 kB | 0.0 kB/s | ETA: 00:00:00 | 100%
gsnull_s.txt | 0 kB | 0.0 kB/s | ETA: 00:00:00 | 100%
gsw_ldef.txt | 0 kB | 0.5 kB/s | ETA: 00:00:00 | 100%
GS_Auto.txt | 1 kB | 1.7 kB/s | ETA: 00:00:00 | 100%
```



## Allow user certificate logon

Enable or disable user Digital Certificate Logon

Users are allowed to use Digital Certificates for logon authentication. Some environments may want to restrict this capability and not allow Digital Certificate Logons. This can be controlled by the registry key `bEnableLogonCertificate`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\bEnableLogonCertificate
```

The default value is `0x01`. (Allow Digital Certificate Logon). The value `0x00` disallows Digital Certificate Logon.

The following is the procedure to change the registry key for enabling or disabling Digital Certificate Logon.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonCertificate
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `bEnableLogonCertificate` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Allow user public key logon

Enable or disable Public Key Logon

Users are allowed to use public key authentication.

Some environments may want to restrict this capability and not allow public key authentication. This can be controlled by the registry key `bEnableLogonPublicKey`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonPublicKey
```

The default value is `0x01`. (Allow public key logon). The value `0x00` disallows public key Logon.

The following is the procedure to change the registry key for enabling or disabling public key Logon.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonPublicKey
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `bEnableLogonPublicKey` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Allow User Name/Password logon

Enable or disable user name/password logon

Users are allowed to use user name/password logon authentication.

Some environments may want to restrict this capability and not allow user name/password authentication.

This can be controlled by the registry key `bEnableLogonPassword`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonPassword
```

The default value is `0x01`. (Allow user name/password logon). The value `0x00` disallows user name/password logon.

The following is the procedure to change the registry key for enabling or disabling user name/password logon.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonPassword
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the bEnableLogonPassword and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## **Allow user GSSAPI logon**

Enable or disable GSSAPI logon authentication

Users are allowed to use GSSAPI logon authentication.

Some environments may want to restrict this capability and not allow GSSAPI authentication.

This can be controlled by the registry key bEnableLogonGSSAPI. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonGSSAPI
```

The default value is 0x01. (Allow GSSAPI logon). The value 0x00 disallows GSSAPI logon.

The following is the procedure to change the registry key for enabling or disabling user name/password logon.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableLogonGSSAPI
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `bEnableLogonGSSAPI` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Listen on IPv4 interfaces

Specify the address SSH Shield listens on for IPv4 connections.

This can be controlled by the registry key `szBindIPv4Address`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szBindIPv4Address
```

The default value is the empty string. (Listen on all IPv4 interfaces).

The following is the procedure to change the registry key to specify the address to listen for IPv4 connections.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szBindIPv4Address
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szBindIPv4Address` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Listen on IPv6 interfaces

Specify the address SSH Shield listens on for IPv6 connections.

This can be controlled by the registry key `szBindIPv6Address`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szBindIPv6Address
```

The default value is the empty string. (Listen on all IPv6 interfaces).

The following is the procedure to change the registry key to specify the address to listen for IPv6 connections.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szBindIPv6Address
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szBindIPv6Address` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Allow use of the “Exec” channel

Allow/Disallow users using the “exec” channel as defined in the SSH Connection Protocol. The ‘Exec channel’ allows to use ssh ‘exec’ capable clients in scripts.

This functionality may be accessed as shown in this example

```
ssh luke@gsw2003 cmd /c dir
```

You can Allow/Disallow this capability by modifying the following registry key.

This configuration is contained in the registry key `bAllowServiceExecute` . The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\bAllowServiceExecute
```

The default value is `0x0` (Do not allow access to the “exec” channel). The value `1` enables the use of the “exec” channel.

The following is the procedure to change the registry key for allowing the use of the “exec” channel.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT

4. Click **OK**

5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowServiceExecute
```

6. Select the menu item **Edit** and then click on **Modify**

7. Enter the new value for the `bAllowServiceExecute` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Enable RSA Host Key

Enable or disable the use of the RSA host key for server-client-authentication.

This can be controlled by the registry key `bAllowRSAHostKey`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\bAllowRSAHostKey
```

The default value is the `0x01` . (RSA Host Key is enabled to be used for server-to-client authentication). `0x00` will disable its use for server-to-client authentication.

The following is the procedure to change the registry key to enable or disable the use of the RSA Host Key.

1. Click the **Start** button at the bottom left corner of your screen.

2. Click **RUN**

3. Type REGEDIT

4. Click **OK**

5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowRSAHostKey
```

6. Select the menu item **Edit** and then click on **Modify**

7. Enter the new value for the `bAllowRSAHostKey` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Enable DSA Host Key

Enable or disable the use of the DSA host key for server-client-authentication.

This can be controlled by the registry key `bAllowDSAHostKey`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\bAllowDSAHostKey
```

The default value is the `0x01`. (DSA Host Key is enabled to be used for server-to-client authentication). `0x00` will disable its use for server-to-client authentication.

The following is the procedure to change the registry key to enable or disable the use of the DSA Host Key.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bAllowDSAHostKey
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `bAllowDSAHostKey` and click **OK**  
*The new value will take effect when the GSW SSHD service is restarted*

## Enable ECDSA Host Key

Enable or disable the use of the ECDSA host key for server-client-authentication.

This can be controlled by the registry key `bAllowECDSAHostKey`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\bAllowECDSAHostKey
```

The default value is the `0x01`. (ECDSA Host Key is enabled to be used for server-to-client authentication). `0x00` will disable its use for server-to-client authentication.

The following is the procedure to change the registry key to enable or disable the use of the ECDSA Host Key.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

HKEY\_LOCAL\_MACHINE\SOFTWARE Wow6432Node\Georgia SoftWorks\GSW\_SSHD\Parameters\bAllowECDSAHostKey

6. Select the menu item **Edit** and then click on **Modify**
  7. Enter the new value for the bAllowECDSAHostKey and click **OK**
- The new value will take effect when the GSW SSHD service is restarted*



## Encryption algorithm categories and lists overview

The main categories of Encryption Algorithms are Host Key Algorithms, Key Exchange Algorithms, Ciphers and MACs. There are also public key and compression algorithms. Each category has a registry key where the value contains the list of algorithms that can be used. In some cases special values enable predefined lists of algorithms.

For each category there may be up to four algorithm lists that are preconfigured and ready to use.

- GSW Default
- FIPS 140-2 Disabled
- FIPS 140-2 Enabled
- All Available Algorithms

You may also create your own custom list of algorithms from the available algorithms. This gives you the flexibility to restrict connections to only certain strength algorithms or to allow only specific legacy algorithms or any combination thereof.

GSW determines the best available algorithms for each category at each release. The list of algorithms setup at installation for each category is called the "GSW Default Algorithms list". No configuration is required to use the GSW Default Algorithms.

The FIPS 140-2 Enabled list is a list of algorithms that does not include any algorithms that are not supported by FIPS 140-2. When you Enable FIPS 140-2 and restart the SSH service, the FIPS 140-2 Enabled list is activated.

The FIPS 140-2 Disabled is a list of all the algorithms available.

In some cases there is no difference in the algorithms available for FIPS 140-2 Enabled and Disabled. We call this list "All Available Algorithms". Not surprisingly, it is also used to refer to all the available algorithms.

If the empty string is configured for the value, then you get the complete list of algorithms available based on the FIPS 140-2 setting.

### Specify Key Exchange Algorithms

Specify the Key Exchange algorithms available to the server that are offered to the client. The Key Exchange algorithms are offered to the client in the server's default order unless specified. The default order will vary from release to release to deliver the best blend of security and performance.

Specify the ciphers that the server can offer to the client by modifying the registry key `szKexAlgorithms`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szKexAlgorithms
```

The following is the default list for Key Exchange Algorithms .

```
curve25519-sha256,curve25519-sha256@libssh.org,diffie-hellman-group18-sha512,diffie-hellman-group16-sha512,diffie-hellman-group-exchange-sha256,diffie-hellman-group14-sha256,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1,diffie-hellman-group1-sha1
```

The following is the list and order of all Key Exchange Algorithms available.

```
ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,curve25519-sha256,curve25519-sha256@libssh.org,diffie-hellman-group-exchange-sha256,diffie-hellman-group16-sha512,diffie-hellman-group14-sha256,diffie-hellman-group18-sha512,diffie-hellman-group-exchange-sha1,diffie-hellman-group1-sha1,diffie-hellman-group14-sha1
```

The list of Key Exchange Algorithms does not vary based the Enable/Disable value for FIPS 140-2 option.

The following is the procedure to change the registry key to specify the Key Exchange Algorithms available to the client.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szKexAlgorithms
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szKexAlgorithms` and click **OK**  
*The new value will take effect when the GSW SSHD service is restarted*

## Specify Ciphers

Specify the ciphers available to the server that are offered to the client. The ciphers are available to the client in the server's default order unless specified. The default order will vary from release to release to deliver the best blend of security and performance.

Specify the ciphers that the server can offer to the client by modifying the registry key `szCiphers`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szCiphers
```

The following is the default list of ciphers.

```
aes256-gcm@openssh.com, chacha20-poly1305@openssh.com, aes256-ctr, aes192-ctr, 3des-cbc, aes128-ctr, aes128-gcm@openssh.com, aes256-cbc, rijndael256-cbc, rijndael-cbc@lysator.liu.se, aes192-cbc, rijndael192-cbc, aes128-cbc, rijndael128-cbc, cast128-cbc, blowfish-cbc
```

The following is the list and order of all ciphers available with FIPS 140-2 disabled

```
aes128-cbc, chacha20-poly1305@openssh.com, aes128-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com, 3des-cbc, blowfish-cbc, aes192-cbc, aes192-ctr, aes256-cbc, aes256-ctr, rijndael128-cbc, rijndael192-cbc, rijndael256-cbc, rijndael-cbc@lysator.liu.se, cast128-cbc
```

The following is the list and order of ciphers available with the FIPS 140-2 option enabled.

```
aes128-cbc, aes128-ctr, 3des-cbc, aes192-cbc, aes192-ctr, aes256-cbc, aes256-ctr, aes128-gcm@openssh.com, aes256-gcm@openssh.com, rijndael128-cbc, rijndael192-cbc, rijndael256-cbc, rijndael-cbc@lysator.liu.se
```

The registry parameter `bDisableFIPS` must be set to 1 to use algorithms which are not on the FIPS list. The default value of this parameter is 0.

The following is the procedure to change the registry key to specify the Ciphers available to the client.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szCiphers
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szCiphers` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

### Specify Message Authentication Code Algorithms (MACs)

Specify the Message Authentication Code algorithms available to the server that are offered to the client. The Message Authentication Code algorithms are offered to the client in the server's default order unless specified. The default order will vary from release to release to deliver the best blend of security and performance.

Specify the Message Authentication Algorithms that the server can offer to the client by modifying the registry key `sZMACs`. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\sZMACs
```

The following is the default value for Message Authentication Code algorithms.

```
hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512,hmac-sha2-256,hmac-sha1-etm@openssh.com,hmac-sha1,hmac-sha1-96,hmac-md5,none
```

The following is the list and order of all algorithms available with the FIPS 140-2 option disabled.

```
hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-96,hmac-md5,none
```

The following is the list and order of algorithms available with the FIPS 140-2 option enabled.

```
hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-96
```

The registry parameter `bDisableFIPS` must be set to 1 to use algorithms which are not on the FIPS list. The default value of this parameter is 0.

The following is the procedure to change the registry key to specify the Message Authentication Code algorithms available to the client.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\sZMACs
```

6. Select the menu item **Edit** and then click on **Modify**

7. Enter the new value for the `sZMACs` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted*

## Algorithm selection for GSW Desktop SSH Client

SSH clients provide a list of Host Key, Key Exchange, Ciphers and MAC algorithms to the SSH Server. The SSH Server goes through each list from the client and for each algorithm chooses the first match from lists that the server supports. If no match is found for any of the algorithms then the connection is refused.

GSW Desktop SSH client provides command line arguments to specify the algorithms and order that are presented to the SSH Server. Customers concerned about achieving high level of security should use the **command line arguments to specify *safe* algorithms as noted in different algorithm tables starting on page 106. *It is recommended to only use the safe algorithms, and not offer unsafe algorithms.***

If you do not specify command lines arguments for the Desktop client, then defaults are used and noted in each section. The defaults are the strongest possible algorithms regardless of FIPs setting

For more detail on Command Line Options for the SSH Clients please see the section *Georgia SoftWorks Desktop Client Command line options – Description* in the GSW UTS Manual.

### Host Key Algorithms List

GSW Desktop SSH Host Key Algorithms (same list with FIPS 140-2 Disabled or Enabled)
<ul style="list-style-type: none"><li>• <b>rsa-sha2-512</b></li><li>• <b>rsa-sha2-256</b></li><li>• ssh-rsa</li><li>• ssh-dss</li><li>• ecdsa-sha2-nistp521</li><li>• ecdsa-sha2-nistp384</li><li>• ecdsa-sha2-nistp256</li></ul>

Table 3: GSW Desktop Host Key Algorithms

If you do not specify the Host Key Algorithms then the default is:

### Default Host Key Algorithms

- rsa-sha2-512

See the master Host Key Algorithm list for all GSW SSH products on page 106. Safe algorithms are in green and unSafe are in red.

## Key Exchange Algorithm List

GSW Desktop SSH Key Exchange Algorithms (same list with FIPS 140-2 Disabled or Enabled)
<ul style="list-style-type: none"><li>• ecdh-sha2-nistp256</li><li>• ecdh-sha2-nistp384</li><li>• ecdh-sha2-nistp521</li><li>• <b>curve25519-sha256</b></li><li>• <b>curve25519-sha256@libssh.org</b></li><li>• diffie-hellman-group-exchange-sha256</li><li>• <b>diffie-hellman-group16-sha512</b></li><li>• <b>diffie-hellman-group14-sha256</b></li><li>• <b>diffie-hellman-group18-sha512</b></li><li>• diffie-hellman-group-exchange-sha1</li><li>• diffie-hellman-group1-sha1</li><li>• diffie-hellman-group14-sha1</li></ul>

Table 4: GSW Desktop Key Exchange Algorithms

If you do not specify the Key Exchange Algorithms then the defaults are listed below:

### Default Key Exchange Algorithms (in order):

- diffie-hellman-group18-sha512
- curve25519-sha256

See the Master Key Exchange algorithm list for all GSW SSH products on page 106. Safe algorithms are in green and unSafe are in red.

### Ciphers List

Note: There are different lists for Ciphers depending if you have FIPs 140-2 enabled.

GSW Desktop SSH Client Ciphers	
SSH with FIPS 140-2 Disabled	with FIPS 140-2 Enabled
<ul style="list-style-type: none"> <li>• aes128-ctr<sup>5</sup></li> <li>• aes128-gcm@openssh.com</li> <li>• aes128-cbc</li> <li>• chacha20-poly1305@openssh.com</li> <li>• 3des-cbc</li> <li>• blowfish-cbc</li> <li>• aes192-ctr<sup>6</sup></li> <li>• aes192-cbc</li> <li>• aes256-gcm@openssh.com</li> <li>• aes256-ctr<sup>7</sup></li> <li>• aes256-cbc</li> <li>• rijndael128-cbc</li> <li>• rijndael192-cbc</li> <li>• rijndael256-cbc</li> <li>• rijndael-cbc@lysator.liu.se</li> <li>• cast128-cbc</li> </ul>	<ul style="list-style-type: none"> <li>• aes128-ctr<sup>5</sup></li> <li>• aes128-gcm@openssh.com</li> <li>• aes128-cbc</li> <li>• 3des-cbc</li> <li>• aes192-ctr<sup>6</sup></li> <li>• aes192-cbc</li> <li>• aes256-ctr<sup>7</sup></li> <li>• aes256-gcm@openssh.com</li> <li>• aes256-cbc</li> <li>• rijndael128-cbc</li> <li>• rijndael192-cbc</li> <li>• rijndael256-cbc</li> <li>• rijndael-cbc@lysator.liu.se</li> </ul>

Table 5: GSW Desktop Ciphers - SSH and with FIPS 140-2

If you do not specify the Ciphers then the defaults are:

#### Default Ciphers (in order):

- aes256-gcm@openssh.com
- chacha20-poly1305@openssh.com
- aes256-ctr

See the master Cipher list for all GSW SSH products on page 107. Safe algorithms are in green and unSafe are in red.

<sup>5</sup> aes128-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>6</sup> aes192-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>7</sup> aes256-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

**Message Authentication Code (MACs) list**

Note: There are different lists for Message Authentication Code algorithms depending if you have FIPs 140-2 enabled.

GSW Desktop SSH MACs	
SSH with FIPS 140-2 Disabled	with <b>FIPS 140-2 Enabled</b>
<ul style="list-style-type: none"> <li>• <b>hmac-sha2-256-etm@openssh.com</b></li> <li>• <b>hmac-sha2-512-etm@openssh.com</b></li> <li>• hmac-sha1-etm@openssh.com</li> <li>• hmac-sha2-256</li> <li>• hmac-sha2-512</li> <li>• hmac-sha1</li> <li>• hmac-sha1-96</li> <li>• hmac-md5</li> <li>• none</li> </ul>	<ul style="list-style-type: none"> <li>• <b>hmac-sha2-256-etm@openssh.com</b></li> <li>• <b>hmac-sha2-512-etm@openssh.com</b></li> <li>• hmac-sha1-etm@openssh.com</li> <li>• hmac-sha2-256</li> <li>• hmac-sha2-512</li> <li>• hmac-sha1</li> <li>• hmac-sha1-96</li> </ul>

Table 6: GSW Desktop MACs both SSH and FIPS 140-2

If you do not specify the MACs then the default is:

**Default MACs:**

- hmac-sha2-512-etm@openssh.com

See the master Message Authentication Code (MACs) list for all GSW SSH products on page 108. Safe algorithms are in green and unSafe are in red.



### SSH Desktop Client Command line Syntax

To manually specify the SSH encryption parameters for the GSW SSH Desktop Client, use the long style arguments on the SSH command line.

Note: Long style arguments have the syntax using the plus "+" sign followed by the argument name, then an equals sign "=" and then the list of algorithms comma separated. These are the algorithms that can be configured on the SSH command line.

```
+HostKeyAlgorithms=<list_of_host_key_algorithms>
```

```
+KexAlgorithms=<list_of_kex_algorithms>
```

```
+Ciphers=<list_of_ciphers>
```

```
+Macs=<list_of_macs>
```

#### EXAMPLE:SSH DESKTOP CLIENT COMMAND LINE ARGUMENTS FOR ALGORITHMS

When I connect to the SSH Server I want the high security and want to use the following algorithms.

```
Host Key Algorithm:      rsa-sha2-512
Key Exchange Algorithms: curve25519-sha256@libssh.org
Ciphers:                 aes256-ctr
MACs:                   hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com
```

A typical command line would look as follows.

```
c:\gs_uts\gs_ssh -ujohndoe -d. -pmostsecure -hsoloman +Ciphers=aes256-ctr
+KexAlgorithms=curve25519-sha256@libssh.org +HostKeyAlgorithms=rsa-sha2-512
+Macs=hmac-sha2-512-etm@openssh.com,hmac-sha2-256-etm@openssh.com
```

Please note the above command is one a single line, so don't be confused by the line wraps.

Please note that the other command line parameters in this example, `-ujohndoe` `-d.` `-pmostsecure` and

`-hsoloman` are just examples for a particular system. The emphasis of this example is to show the algorithms selected.

## Change the SSH Port Number

The default port number is port 22. You can change the port number to the port of your choice.

**Important:** Be sure that you also change the port number on the SSH clients to the same port number configured on the SSH Server.

In the event you want to change the SSH port on the server you can do so by changing the following registry key.

This configuration is contained in the registry key `usGSWSSHDPort` which is a number. The key is:

### For 64-bit

`HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\GeorgiaSoftWorks\GSW_SSHD\Parameters\usGSWSSHDPort`

### For 32-bit

`HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters \usGSWSSHDPort`

The default value is 22.

This following is a procedure to change the registry key for the SSH port number.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type `REGEDIT`
4. Click **OK**
5. Select Registry Key:

**For 64-bit** `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters`

**For 32-bit** `HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters`

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `usGSWSSHDPort` and click **OK**
8. *The new value will take effect when the GSW SSHD service is restarted.*

## Location of SSH Server RSA Private Key

The SSH Server RSA Private Key is in an encrypted file and is in the PEM format.

This configuration is contained in the registry key `szServerRSAKeyFile` which is a text string. You can change the location by modifying the registry key.

The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerRSAKeyFile
```

The default value is the installation folder for the GSW SSH Shield.

```
C:\Program Files\Georgia SoftWorks\Georgia SoftWorks SSH Shield\sshd_rsa.key
```

The following is a procedure to change the Location of SSH Server RSA Private Key.

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerRSAKeyFile
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the `szServerRSAKeyFile` and click **OK**

*The new value will take effect when the GSW SSHD service is restarted.*

## Location of SSH Server DSA Private Key

The SSH Server DSA Private Key is in an encrypted file and is in the PEM format.

This configuration is contained in the registry key `szServerDSAKeyFile` which is a text string. You can change the location by modifying the registry key.

The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerDSAKeyFile
```

The default value is the installation folder for the GSW SSH Shield.

```
C:\Program Files\Georgia SoftWorks\Georgia SoftWorks SSH Shield\sshd_dsa.key
```

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerDSAKeyFile
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the Server DSA Key Location and click **OK**

*The new value will take effect when the GSW SSHD service is restarted.*

## Location of SSH Server ECDSA Private Key

The SSH Server Elliptic Curve Cryptography DSA Private Key is in an encrypted file and is in the PEM format.

This configuration is contained in the registry key `szServerECDSAKeyFile` which is a text string. You can change the location by modifying the registry key.

The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerECDSAKeyFile
```

The default value is the installation folder for the GSW SSH Shield.

```
C:\Program Files\Georgia SoftWorks\Georgia SoftWorks SSH Shield\sshd_ecdsa.key
```

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia  
SoftWorks\GSW_SSHD\Parameters\szServerECDSAKeyFile
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the Server DSA Key Location and click **OK**

*The new value will take effect when the GSW SSHD service is restarted.*

---

## Location of Fingerprints for all Host Keys

The file HostFingerPrints.txt in the Georgia SoftWorks SSH Shield installation folder<sup>8</sup> contains key fingerprints for all host keys offered for server-to-client authentication. These key fingerprints may be entered for host fingerprint configuration of the [Georgia SoftWorks Business Tunnel](#).

The file is formatted as shown below:

```
RSA key MD5 fingerprint ..... d5:bb:19:47:87:05:95:16:a6:d2:ac:db:f1:fc:9c:19

RSA key SHA1 fingerprint .....
8e:ae:0e:ee:05:32:a0:20:b7:10:56:8a:26:88:7b:f0:19:53:32:96

RSA key SHA-256 fingerprint ....
2b:f8:96:8b:5d:c9:9a:5b:18:41:ee:f9:e1:14:57:15:6f:49:84:05:55:b5:87:0c:28:1a:a0:3
b:81:5c:d1:2c

DSA key MD5 fingerprint ..... cd:e8:1e:c8:f0:39:87:dc:1b:b5:18:64:69:56:a3:8b

DSA key SHA1 fingerprint .....
5b:f9:8b:2a:9f:d9:15:48:da:37:38:93:90:3a:ba:e4:91:11:91:05

DSA key SHA-256 fingerprint ....
2f:71:bb:07:84:94:a3:67:59:2e:eb:7b:b2:2a:d9:67:63:60:0f:60:75:93:c1:72:f9:3d:d1:d
b:19:12:bb:ec

ECDSA key MD5 fingerprint ..... 8f:6a:6b:f2:11:25:ba:7e:b6:6c:dc:d7:79:26:66:1e

ECDSA key SHA1 fingerprint .....
42:d0:69:23:e0:a2:b5:4b:04:cd:72:dc:2a:c8:1b:3a:49:c9:ec:1b

ECDSA key SHA-256 fingerprint ..
54:c4:e9:4f:4a:a4:d5:a6:dc:ac:8e:ec:b5:4f:8a:d6:82:76:90:d6:cf:04:18:73:55:0e:6d:8
8:36:8c:5c:b9
```

## Internal SSH Activity Logging FLAG for Debugging

In the event that GSW Technical Support requires additional information, you may need to turn on SSH internal activity logging.

You can activate the internal SSH activity logging by modifying the following registry key.

This configuration is contained in the registry key bEnableWODLog which is a flag. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableWODLog
```

The default value is **0**.

---

<sup>8</sup> Usually 'C:\Program Files\Georgia SoftWorks\Georgia SoftWorks SSH Shield

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key

```
HKEY_LOCAL_MACHINE\SOFTWARE Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\bEnableWODLog
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the Enable Activity Logging and click **OK**  
*The new value will take effect when the GSW SSHD service is restarted.*

## Internal SSH Activity Log file location for Debugging

In the event that GSW Technical Support requires additional information, you may need change the SSH internal activity log file location.

You can modify the internal SSH activity log file name and location by modifying the following registry key.

This configuration is contained in the registry key `szWODLogFile` which is a text string. The key is:

```
HKEY_LOCAL_MACHINE\SOFTWARE Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szWODLogFile
```

The default value is the log folder in the GSW UTS Installation directory. Usually this is:

```
C:\GS_UTS\log
```

**NOTE:** `bEnableWODLog` must be set to 1 for the log file to operate.

Note: (you must be on the Windows NT/XP/VISTA/2000+ system that the Georgia SoftWorks SSH Server is installed. However, you may connect to the SSH Registry from a remote location).

1. Click the **Start** button at the bottom left corner of your screen.
2. Click **RUN**
3. Type REGEDIT
4. Click **OK**
5. Select Registry Key

```
HKEY_LOCAL_MACHINE\SOFTWARE Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters\szWODLogFile
```

6. Select the menu item **Edit** and then click on **Modify**
7. Enter the new value for the Enable Log File Name and Location and click **OK**

*The new value will take effect when the GSW SSHD service is restarted.*



## SSH Server Mapping Tool for Certificates and Public Keys

Georgia SoftWorks researched and developed an innovative, easy to use, and secure implementation of Digital Certificates<sup>9</sup>. The result of this effort is the GSW SSH SHIELD Certificate Mapping Tool.

The entire configuration is done through a GUI with wizard style dialogs reminiscent of IIS certificate-to-user account mapping. The solution preserves all of the cryptographic strength of the public key solution, adds convenient, well scaling, certificate-to-user account mapping options.

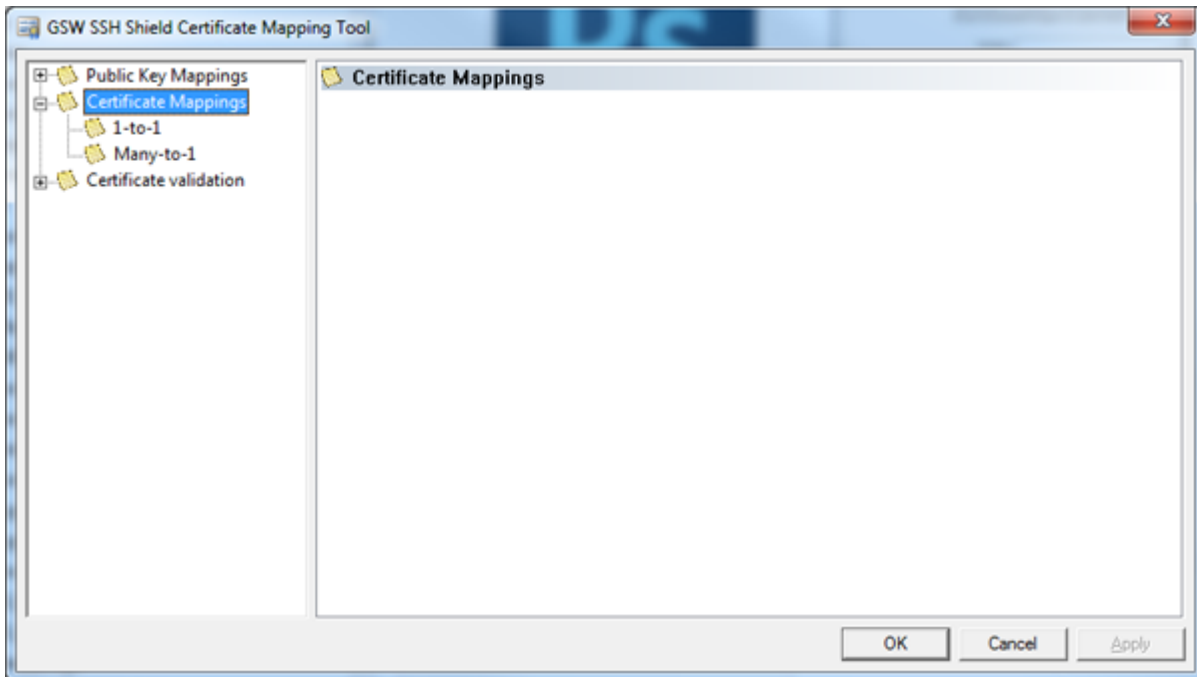


Figure 32 - SSH Certificate Mapping Tool

The overall solution allows authenticating SSH users who log on with a client certificate by mapping the certificates to Windows user accounts. The client certificates are analyzed and used to either deny or grant host access to a connecting session.

There are two methods in which one can map certificates.

---

<sup>9</sup> A Digital Certificate binds a name (or identity) to a public key value and is used in verifying the identity of the certificates owner.

## Certificate One-to-one mapping

'One-to-one' mapping maps a individual client certificate to a individual Windows user account. The SSH-2 server compares certificates from a pre-configured list with the client certificate that is sent by the SSH-2 client. An identical match must occur for the mapping to proceed.

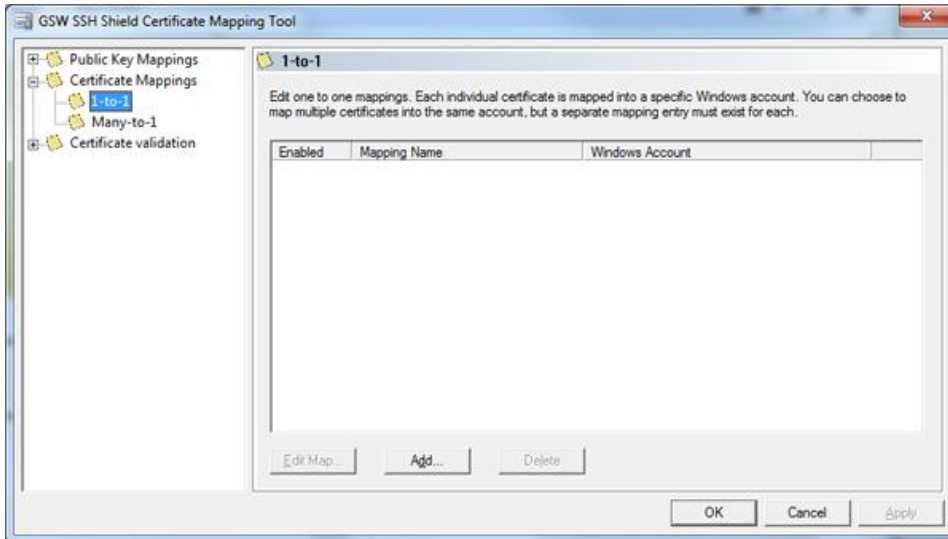


Figure 33 - One-to-one certificate mapping

## Certificate Many-to-one mapping

'Many-to-one' mapping maps multiple certificates to an individual Windows user account. It uses wildcard matching rules to define the certificate criteria for mapping. This type of mapping does not compare the actual client certificate. Instead, it accepts all client certificates that meet specific criteria. If a certificate matches the rules, it is mapped to the indicated user account. Typically one would also select a Certificate Trust List (CTL) to assure the client certificates are truly trustworthy. CTLs make it possible to limit the number of acceptable root CAs which are able to issue certificates to users.

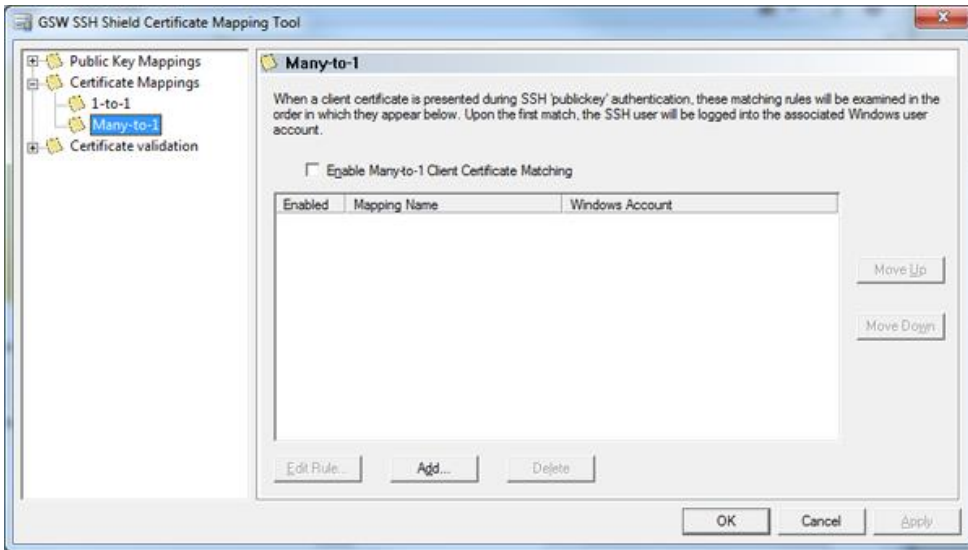


Figure 34 - Many-to-one certificate mapping

### Public Key 1-to-1 mapping

Public Key '1-to-1' mapping provides a very nice method to allow public key to user account mapping.

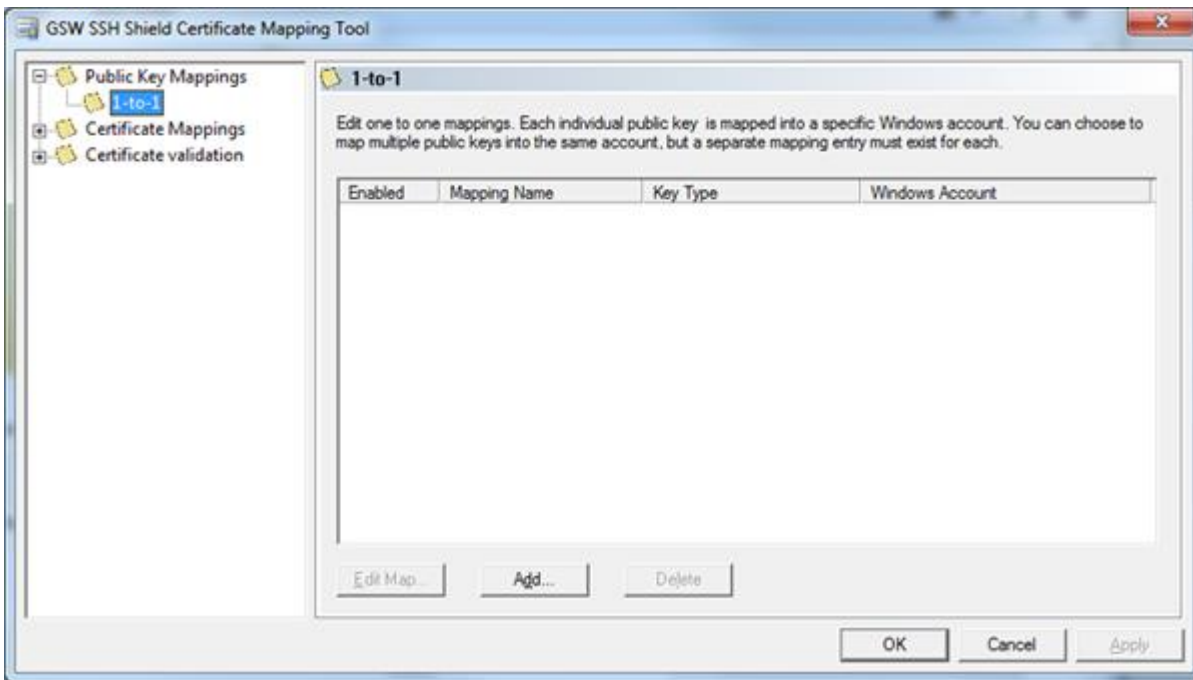


Figure 35 - Public Key Mappings - 1-to-1

## Certification Validation – Certificate Trust List

You can also configure Certificate Trust List (CTL) with the GSW Mapping Tool.

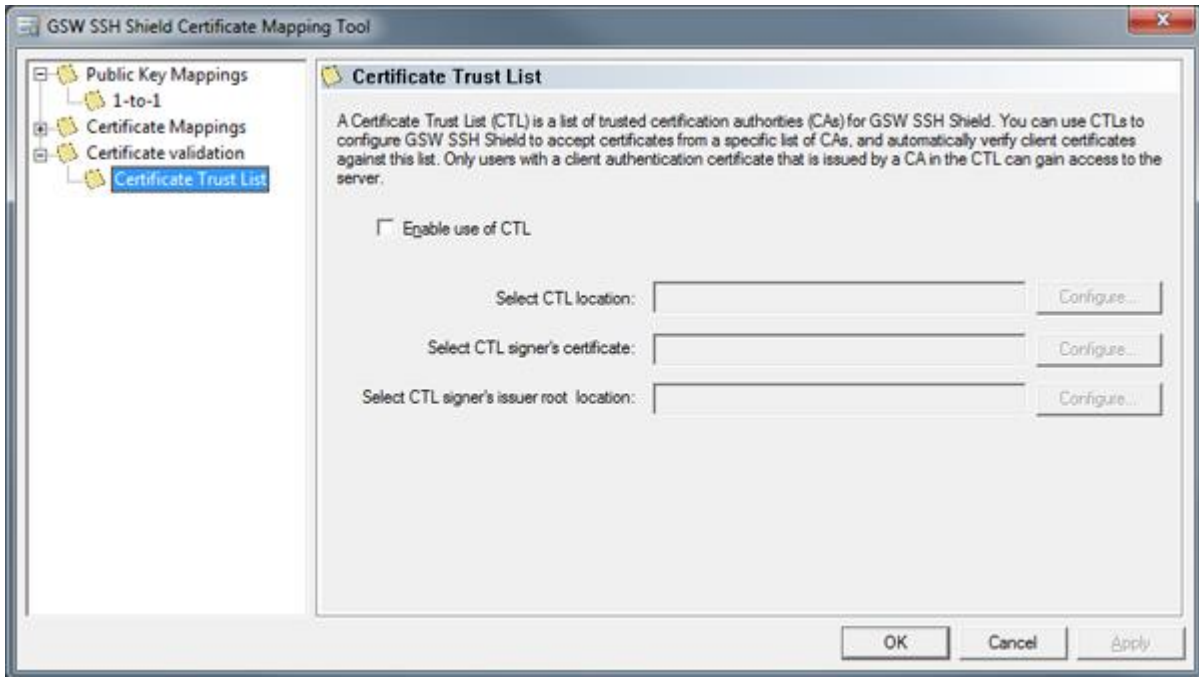


Figure 36 - Certificate Validation Certificate Trust List

## Public/Private Key Creation and Use

### Public / Private Key Introduction

Public key authentication is far more secure than passwords and provides exceptional usability benefits. The cryptographic strength of the recommended and trustworthy algorithms are superior to even the longest passwords. The user does not have to remember a password and Public Key Authentication provides automatic logon capabilities. The benefits are substantial.

When using this type of authentication, the entity (SSH client software) being authenticated has a public key and a private key. The private key is kept on the client, while the public key is stored on the server.

In this example we will generate a Public/Private key pair, install the private key on the client and the public key on the server. We also map the public key to a Windows user account, which provides automatic logon on.

## Creating a Public / Private Key

There are a multitude of tools available to generate a Public/Private key pair, many at no cost. In this example we will use PuTTYgen to create a public and private key.

1. Download [PuTTYgen](#). Click on the PuTTYgen executable.
2. Select the SSH-2 RSA radio button, near the bottom of the window.
3. Create a key set by clicking on the "Generate" button.

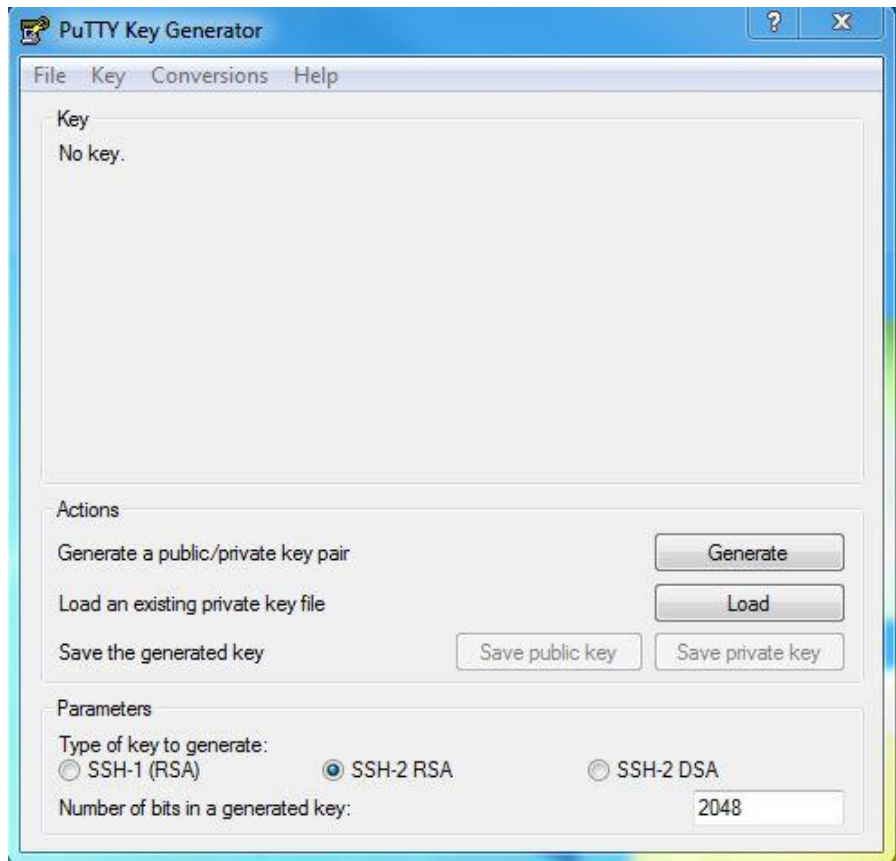


Figure 37: Puttygen Tool for Creating Public/Private Key Pair

4. Randomly move the mouse pointer in the open area under the progress bar, until the progress bar completes.

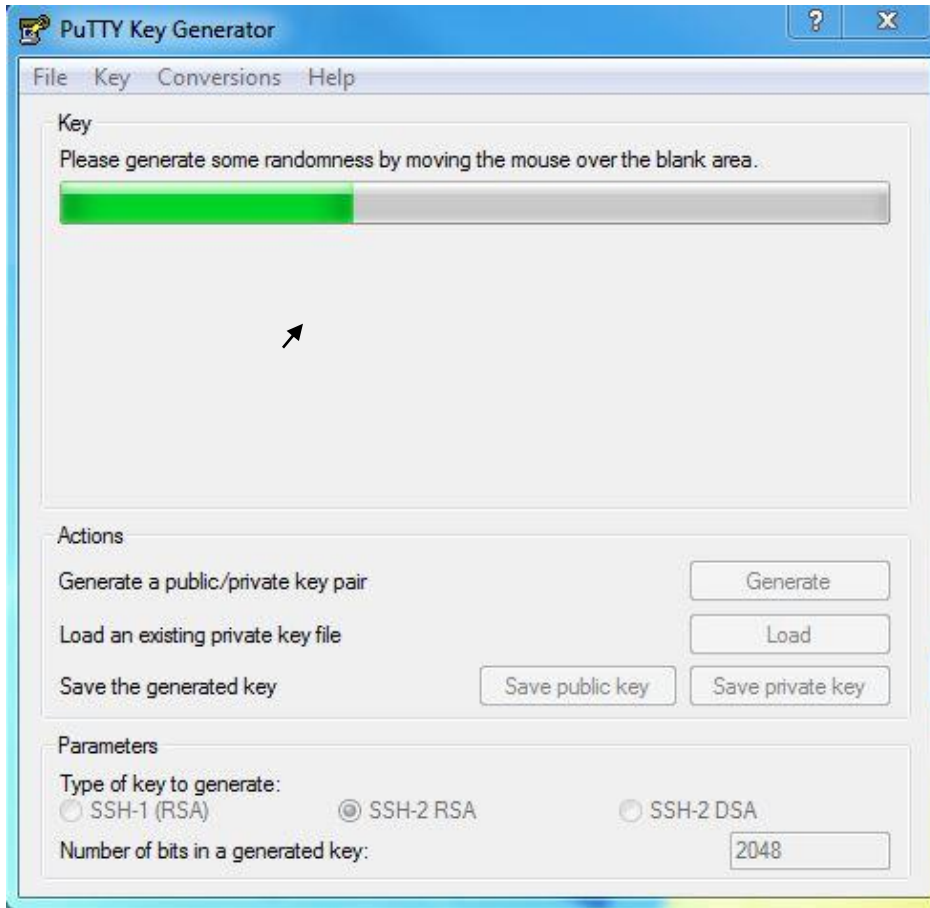


Figure 38: Puttygen Tool Uses Mouse-Movement to Create Randomness for Key Creation

5. Once the progress bar completes, create an optional key Passphrase<sup>10</sup>.



Figure 39: Puttygen with Generated Key

6. Click Save public key and give it a descriptive file name and location to be stored.
7. Click Save private key and give it a descriptive file name and location to be stored. You may now close the PuTTYgen tool.

You have now create the public/private key pair. The next step is to install the public key on the server and the private key on the client.

<sup>10</sup> The optional Passphrase is used when the key is installed on the client (see page 76)



## Install Public Key on the Server and Map to Windows User Account

1. Open the Public key you created with notepad or a simple text editor.
2. Copy the highlighted area as shown below to your clip board. Do not include the leading and trailing description tags.

```
---- BEGIN SSH2 PUBLIC KEY ----Comment:"rsa-key-20160621"
```

```
AAAAB3NzaC1yc2EAAAABJQAAAQEAqqX23vaw7ufsRG2C2RocBfheennpjCLlsgn
```

```
zIZMeEa0iIeaNZxwvmlwHokVzY94kvGkMpHcTqUUVAIxifEvgVDthvrHJWiUzN+7
```

```
UZh60XI4g1N+rGag/L/KMQ2ZydQ+eI7iBLq+Fia45k+/fcGZZmDRjHf+rpROpS72
```

```
i5ic2ayIC4bgSEEvzO6dGLTldjzDQO1JiHI5REyia3NmfiSdRaAMHLEmutYypkV
```

```
tXctAWx+aSZo5vIjJ+k/ByFpzQjwpSiLuXEcn+eidDI7ocw2Ely1oqbVjNcqepoo
```

```
Vuq525YEsLkR6XPvoRxJaHzWRQH0911VhJBRZ5Els+DezmGhCQ==
```

```
---- END SSH2 PUBLIC KEY ----
```

3. On the Georgia SoftWorks UTS server, go to Start > All Programs > Georgia SoftWorks UTS > **Certificate Mapping Tool for GSW SSH Shield**
4. Expand Public Key Mapping, and select 1-to-1. Next click Add.

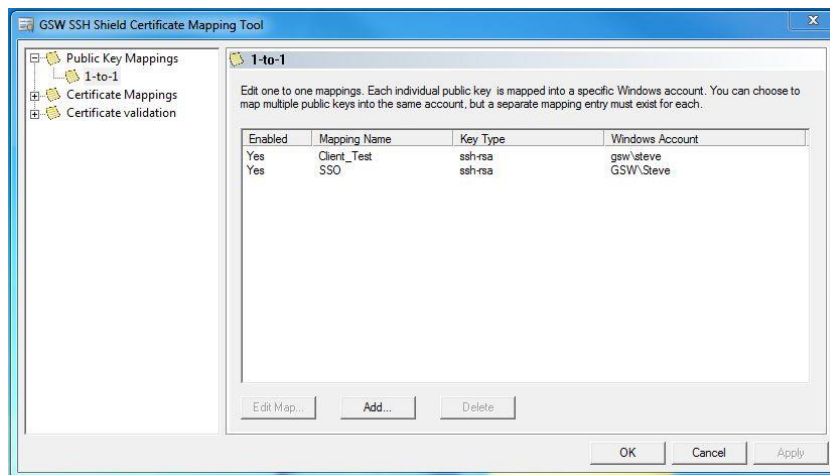


Figure 40: Certificate Mapping Tool

- 5. Click Enable this mapping and paste the clip board data into the Public key area. Fill out the rest of the fields below.

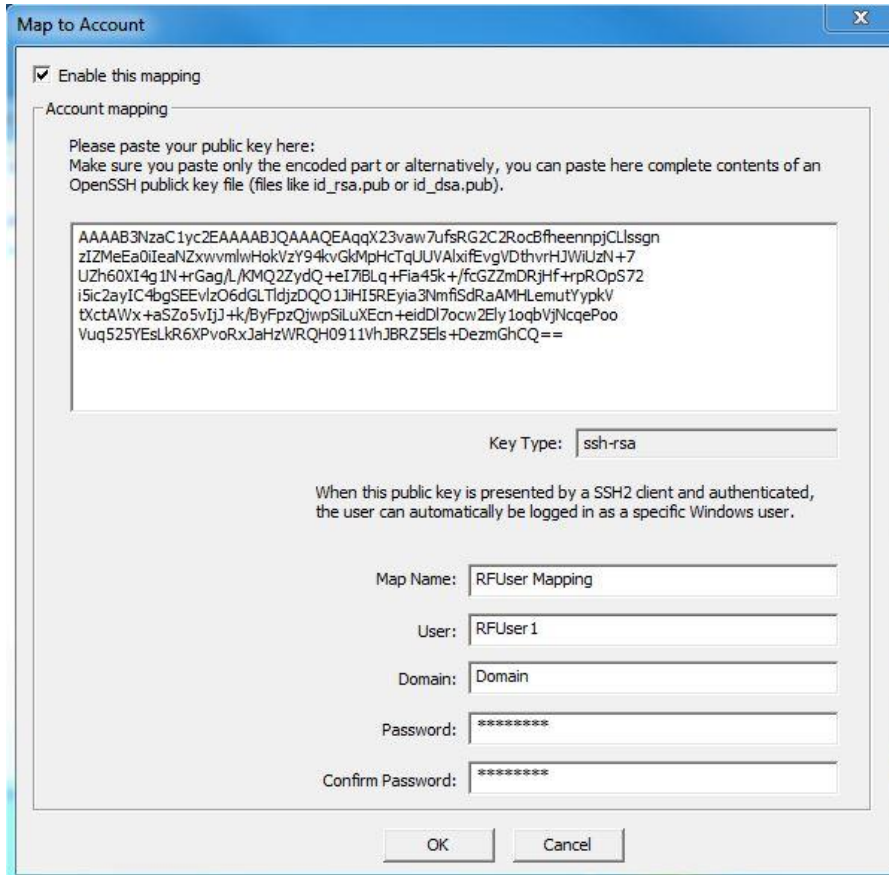


Figure 41: Mapping an Account to a Public Key

- 6. Click OK, and click Apply. You should now see the mapping you created.

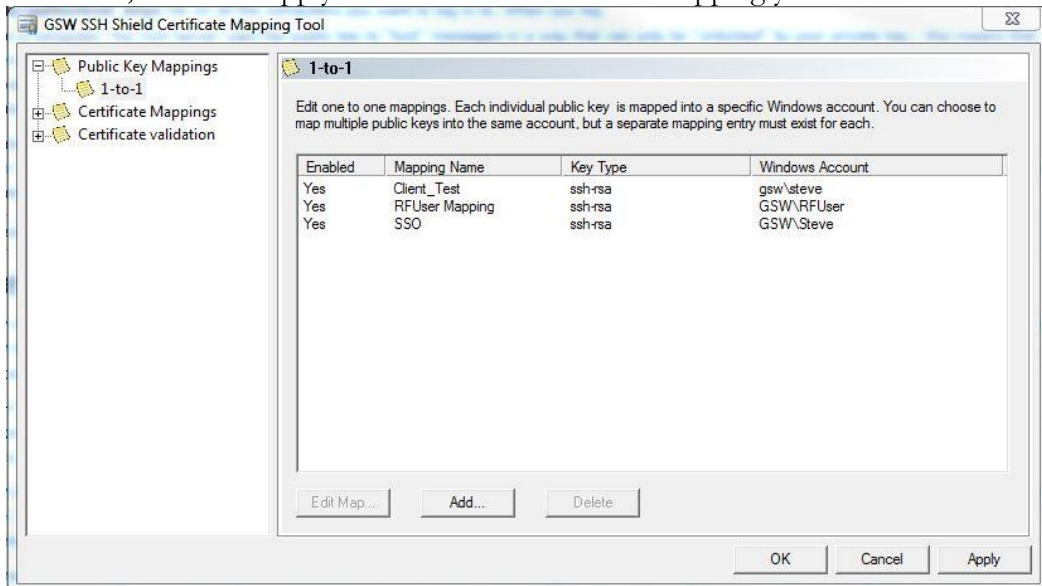


Figure 42: Certificate Mapping Tool – Public Key Mappings

7. The changes will not take effect until the Georgia SoftWorks SSH Shield service is restarted. Click yes to restart SSH or No if you intend to restart later.

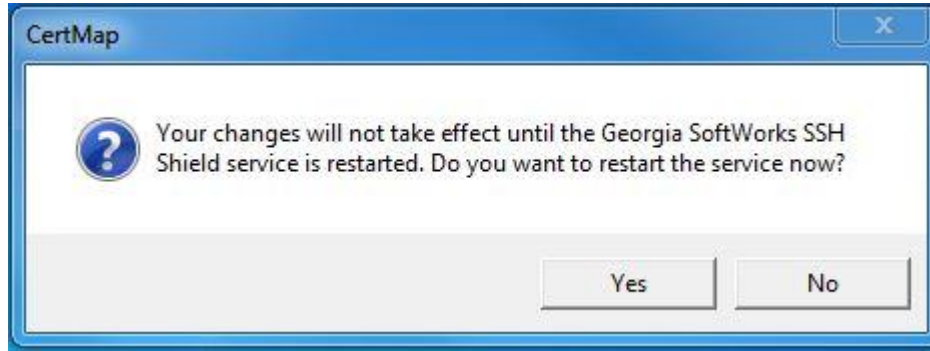


Figure 43: Certificate Mapping Tool Dialogue Box

## Install and Configure Private Key on the Client

Now we will install and configure the Private Key on the GSW client. This will allow the device to connect securely using an SSH encrypted session.

1. Copy the private key generated previously onto the mobile device using [Mobile Device Center](#). You can download it [here](#).
2. Open the GSW client on the device you wish to connect to the GSW server.



Figure 44: Opening the Client on a Device to Set-up Public Key Log-on

3. Select the Session Configuration File you wish to modify. In this case we are selecting `default.gswtc`

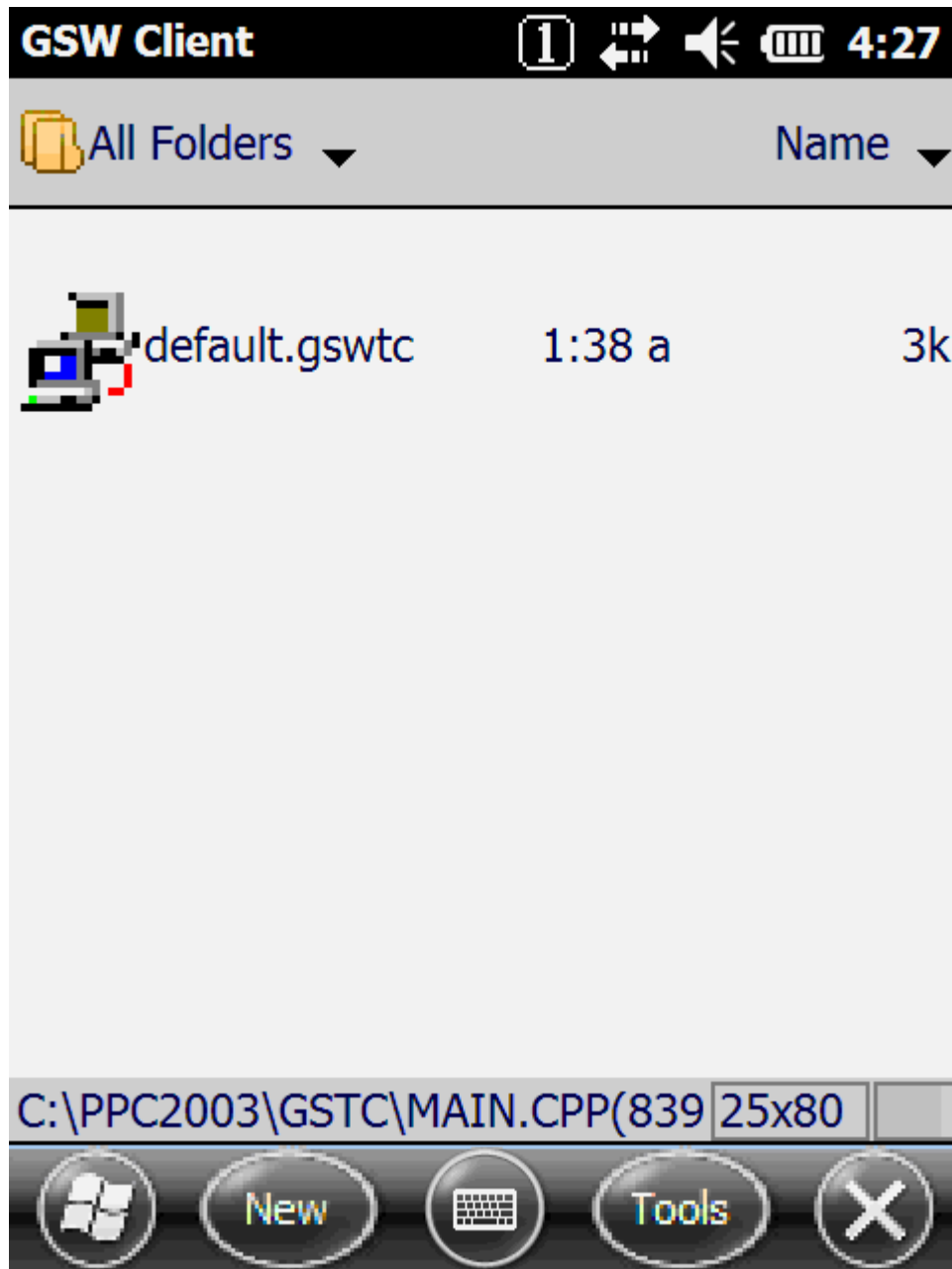


Figure 45: Selecting a Connection profile

- 4. Enter the host information and port, 22 is the default. Select “Options...”

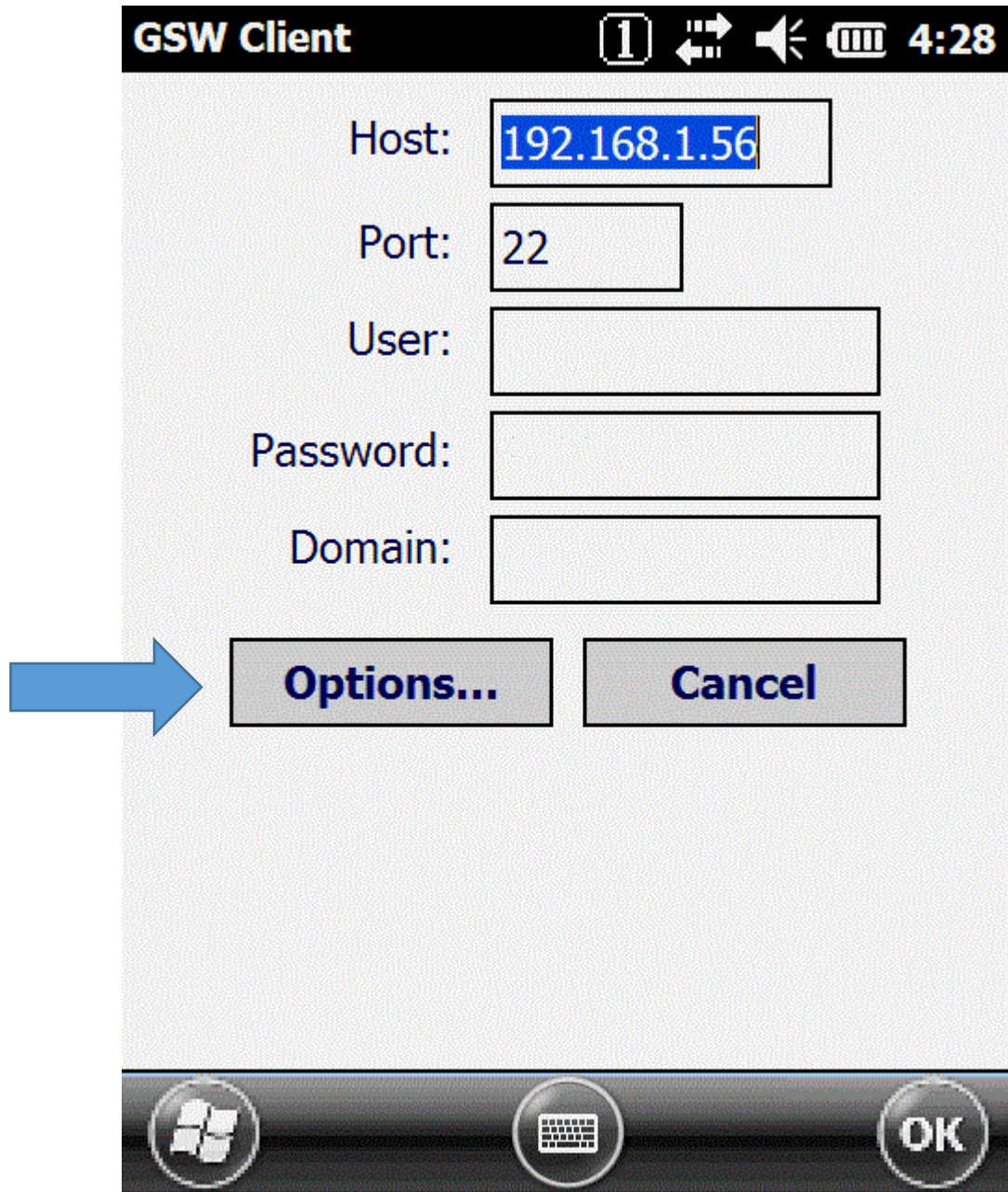


Figure 46: Selecting Options on GSW SSH2 Client to Configure Public Key Log-on

- 5. In the “Encryption” drop down, select SSH2.
- 6. Place a checkmark in the “Public key logon” box.

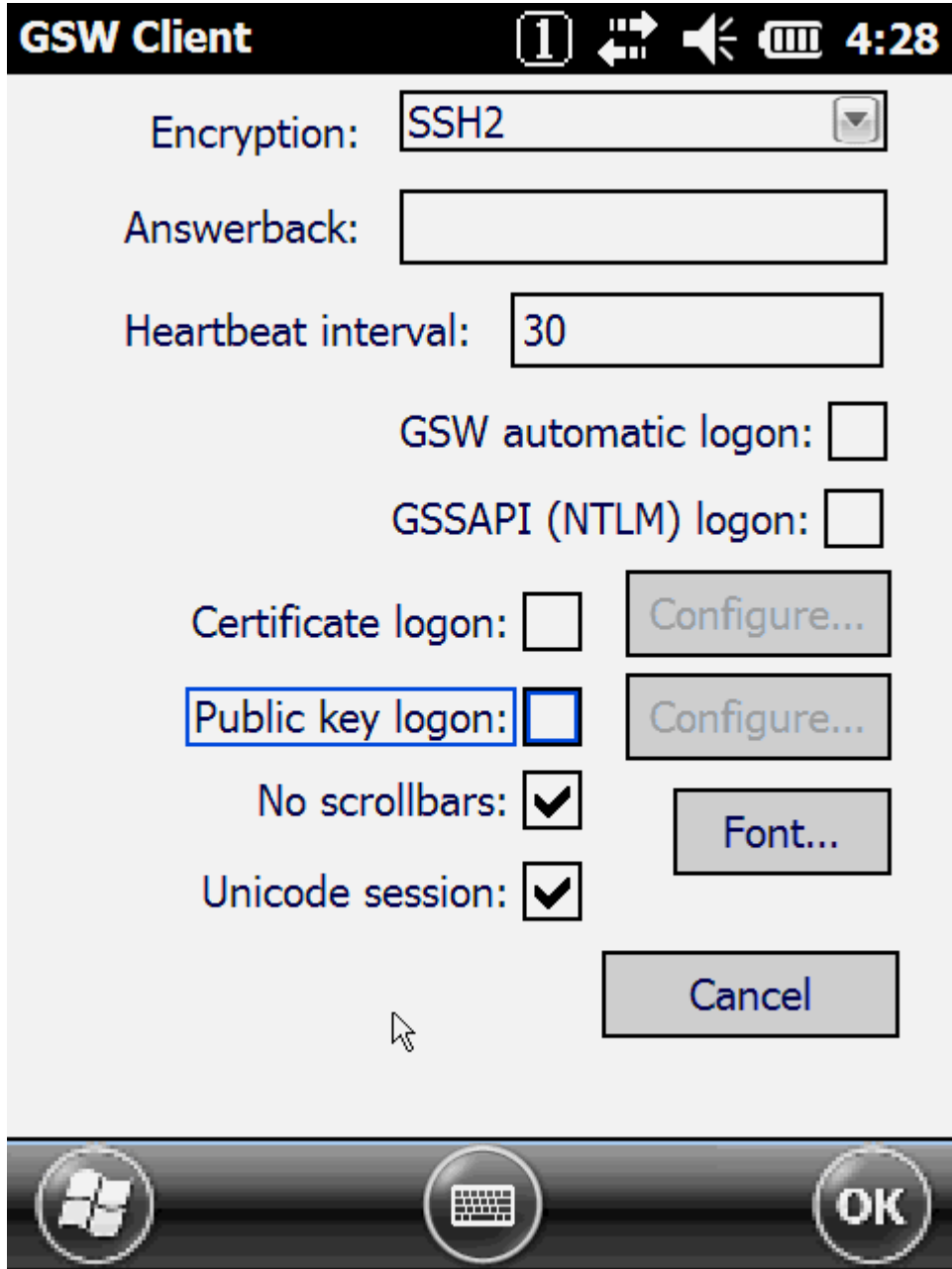


Figure 47: Selecting Public Key Log-on

- 7. This will enable the “Configure” button. Click the “Configure” button.

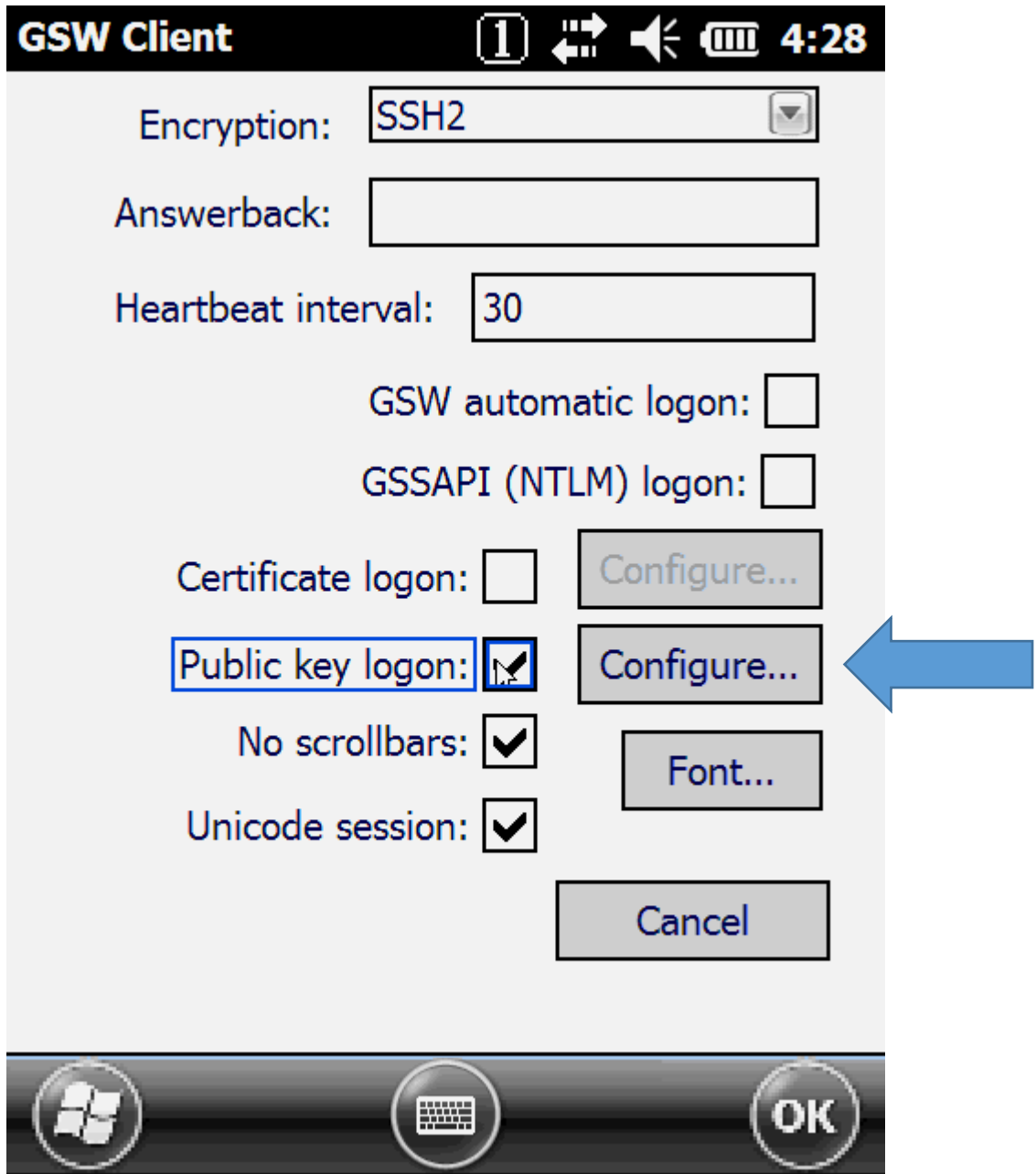


Figure 48: Configuring Public Key Log-on



8. Click the “Browse” button.

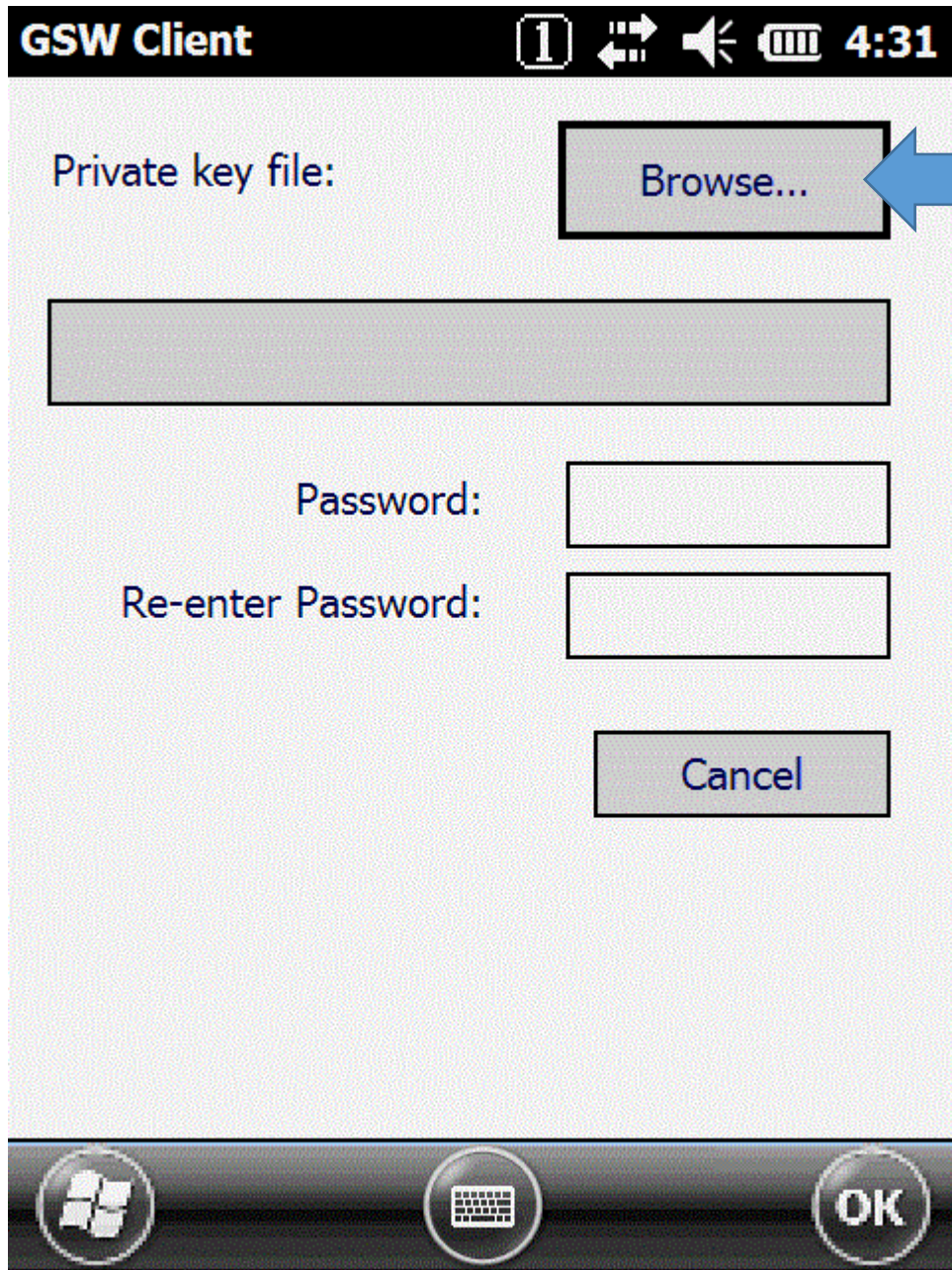


Figure 49: Configuring the Client by Providing a Private Key

- 9. Navigate to the location where you saved the Private Key (\*.ppk) on the device, and select.

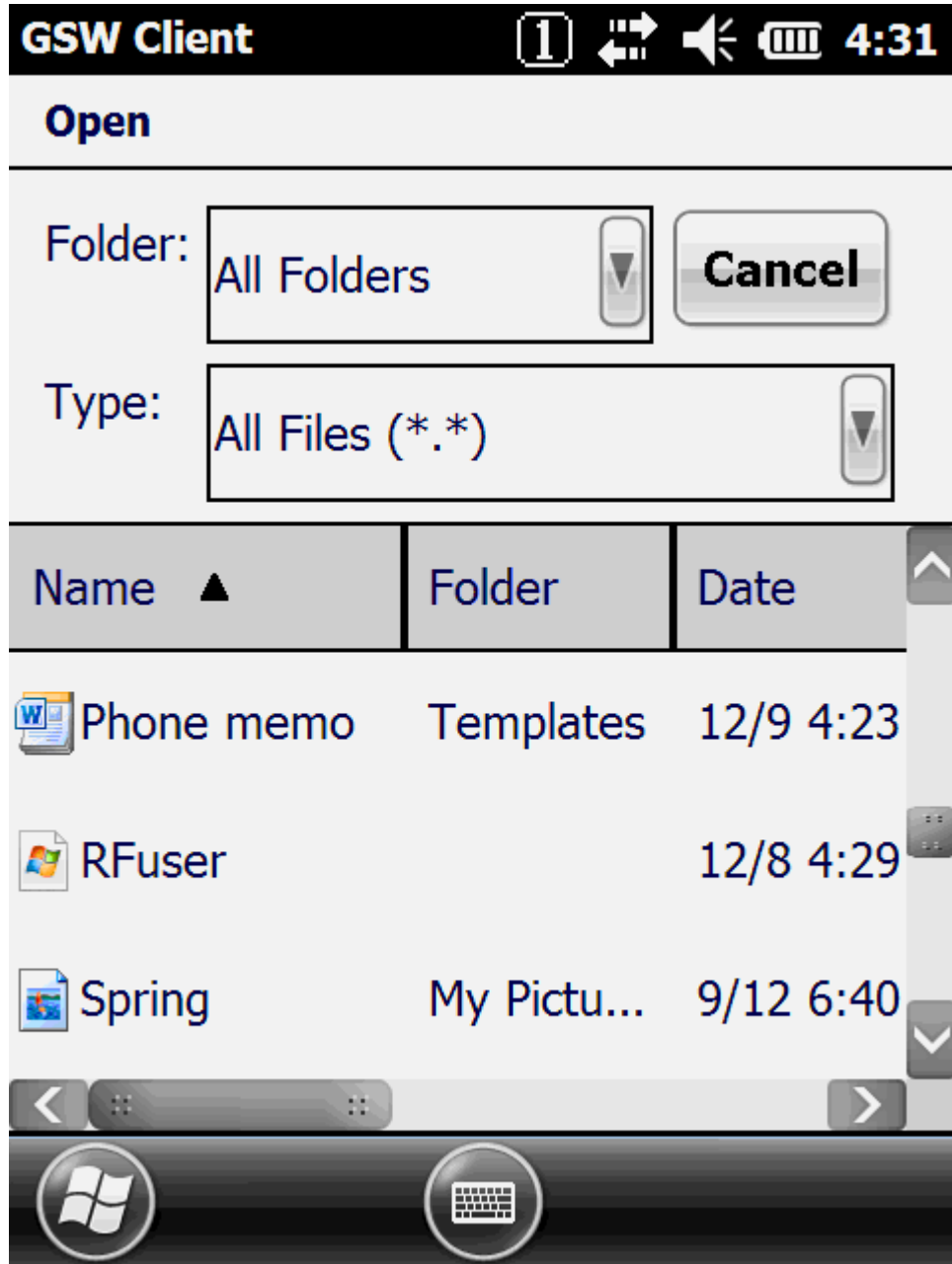


Figure 50: Browsing to Find Private Key

- 10. Enter the passphrase (if created, see page 79), and press the “OK” button at the bottom of the screen. You should see a confirmation that the key was imported successfully. Click “OK” to close the dialog box.

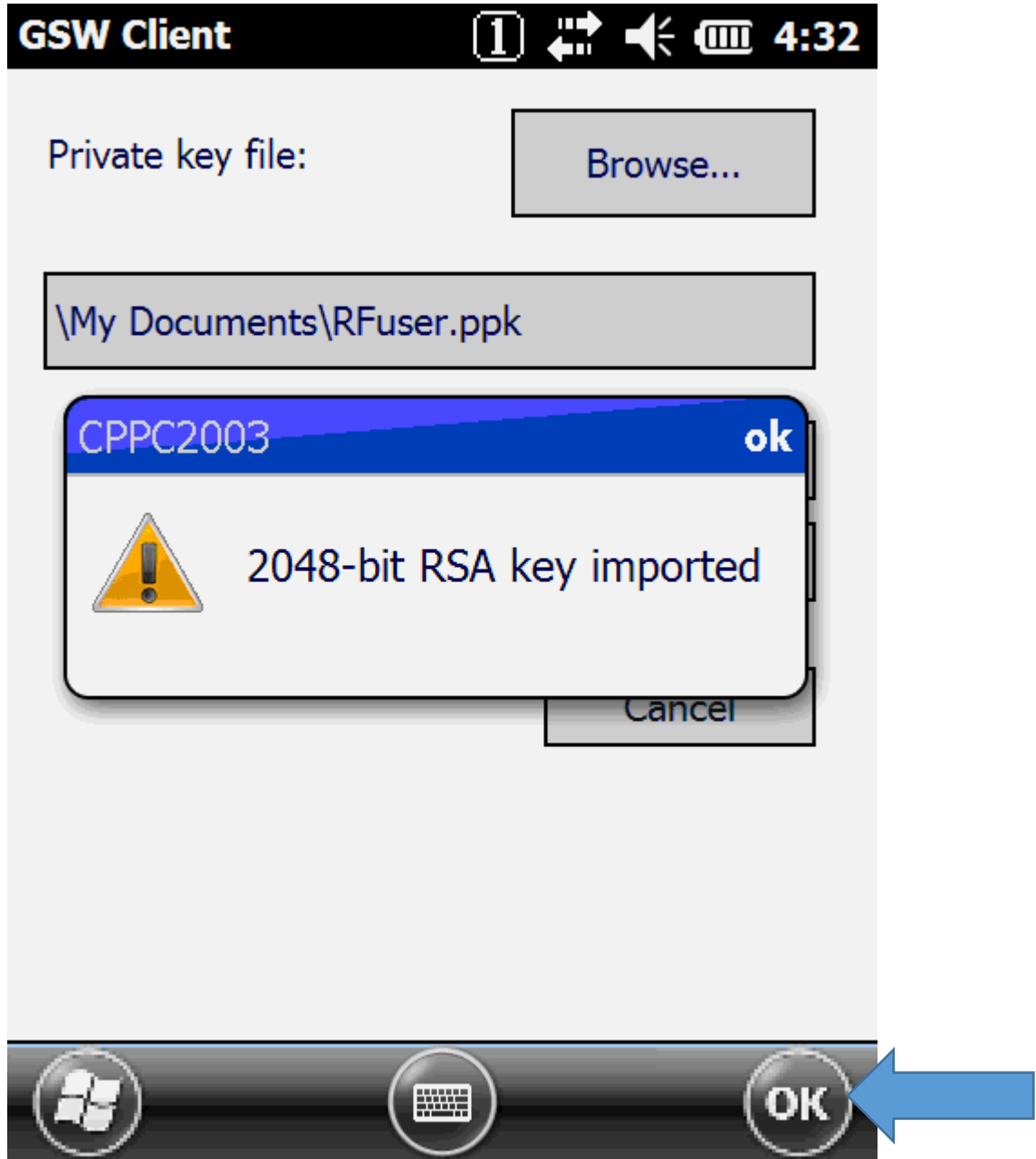


Figure 51: Private Key Imported into Client

11. Launch the GSW client.

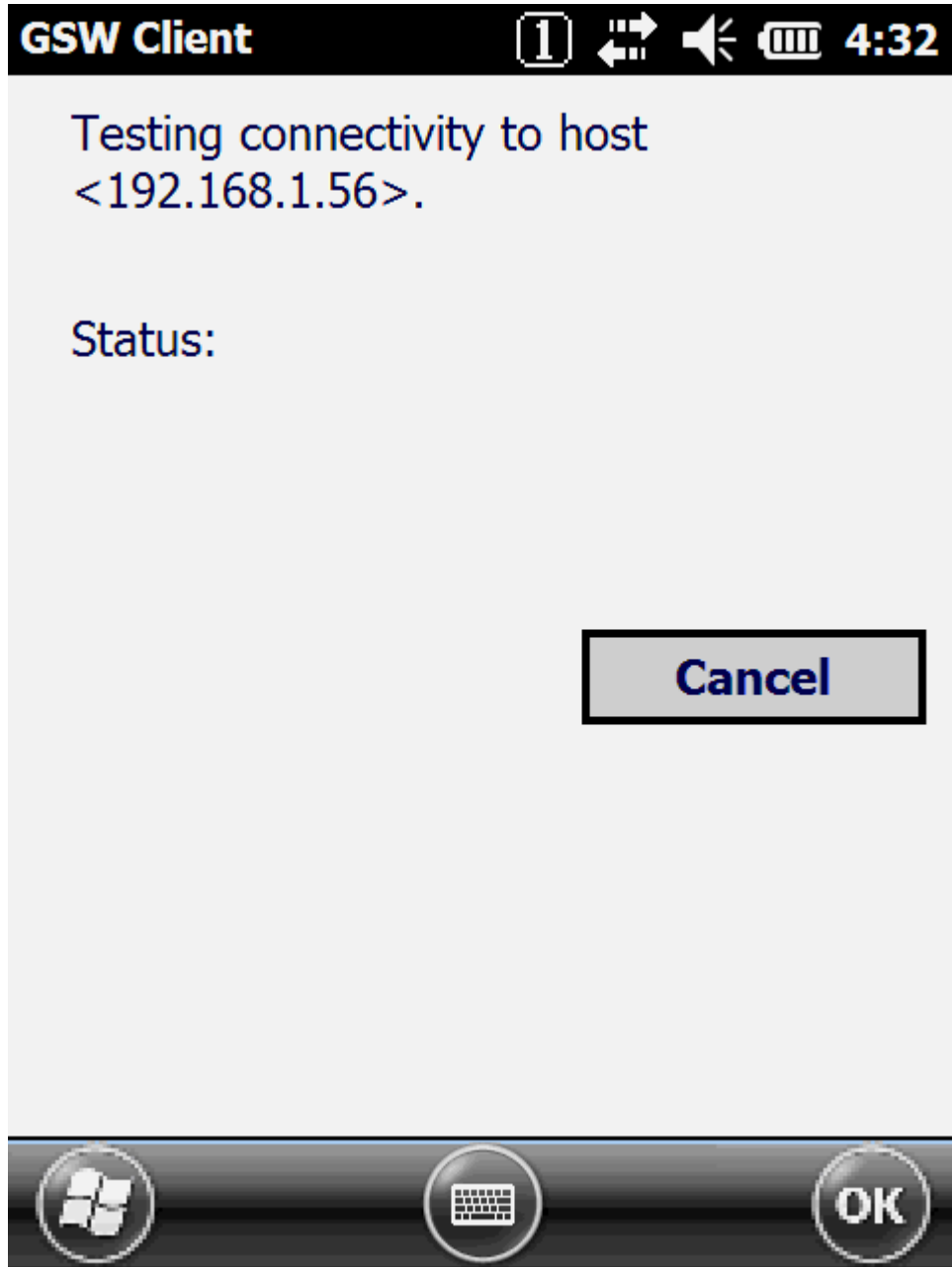


Figure 52: Client Launch Using Public / Private Key Authentication

12. You should see the GSW client connect and begin a session.

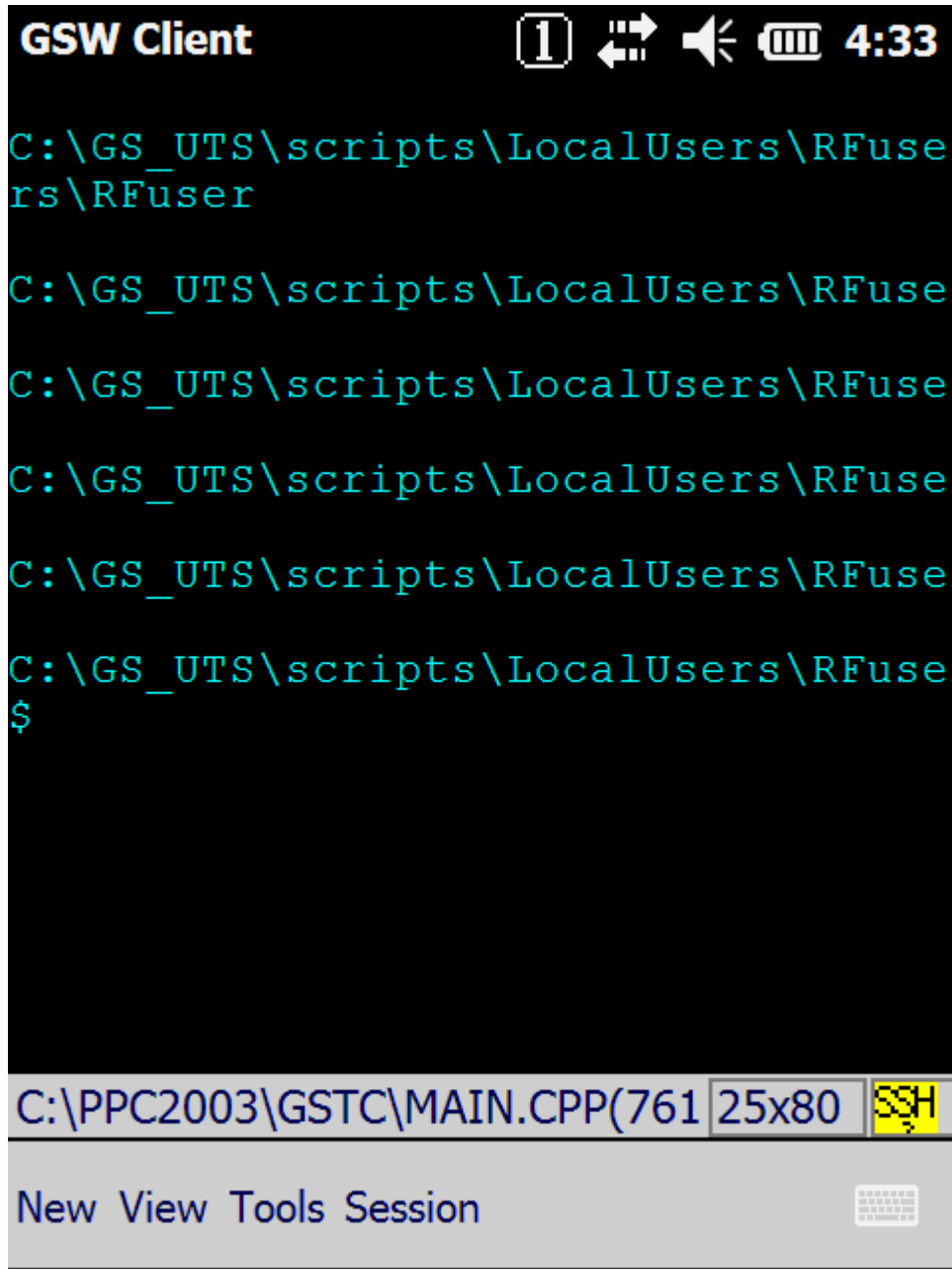


Figure 53: Session Connected via Public / Private Key Authentication

## SSH Clients

In addition to the GSW SSH clients, the Georgia SoftWorks SSH Server is compatible with all SSH compliant third party clients.

### GSW ANDROID SSH CLIENTS

GSW offers GSW ConnectBot, an SSH client for Android. It provides a comprehensive set of features, and offers the strongest encryption available on the commercial market. If you use Android devices and security is a priority, then the GSW ConnectBot is the SSH client you need. It also comes with a license, settings and update server to make administration almost effortless.

Please visit the [GSW ConnectBot web page](#).

### GSW WINDOWS SSH CLIENTS

All the powerful and popular GSW Client options and features described in the GSW UTS are available for the GSW SSH server except where specifically noted. Georgia SoftWorks offers SSH Clients for the following Windows platforms:

Operating System	GSW Widows SSH Client	Method to Launch Client
Window 98/ME	Yes	Program Group Shortcut
Windows NT 4.0	Yes	Program Group Shortcut
Windows 2000	Yes	Program Group Shortcut
Windows XP	Yes	Program Group Shortcut
Windows VISTA	Yes	Program Group Shortcut
Windows 2003	Yes	Program Group Shortcut
Windows 7	Yes	Program Group Shortcut
Windows 8	Yes	Program Group Shortcut
Windows 10	Yes	Program Group Shortcut
Windows 2008/R2	Yes	Program Group Shortcut
Windows 2012/R2	Yes	Program Group Shortcut
Windows 2016	Yes	Program Group Shortcut
Windows 2019	Yes	Program Group Shortcut
Windows CE .NET 4.2+	Yes	Device Desktop Shortcut
Windows Mobile	Yes	Device: Start Programs GSW Telnet and SSH
Pocket PC 2002	No	
Pocket PC 2003	Yes	Device: Start Programs GSW Telnet and SSH
Java Client	No	
Java Applet	No	

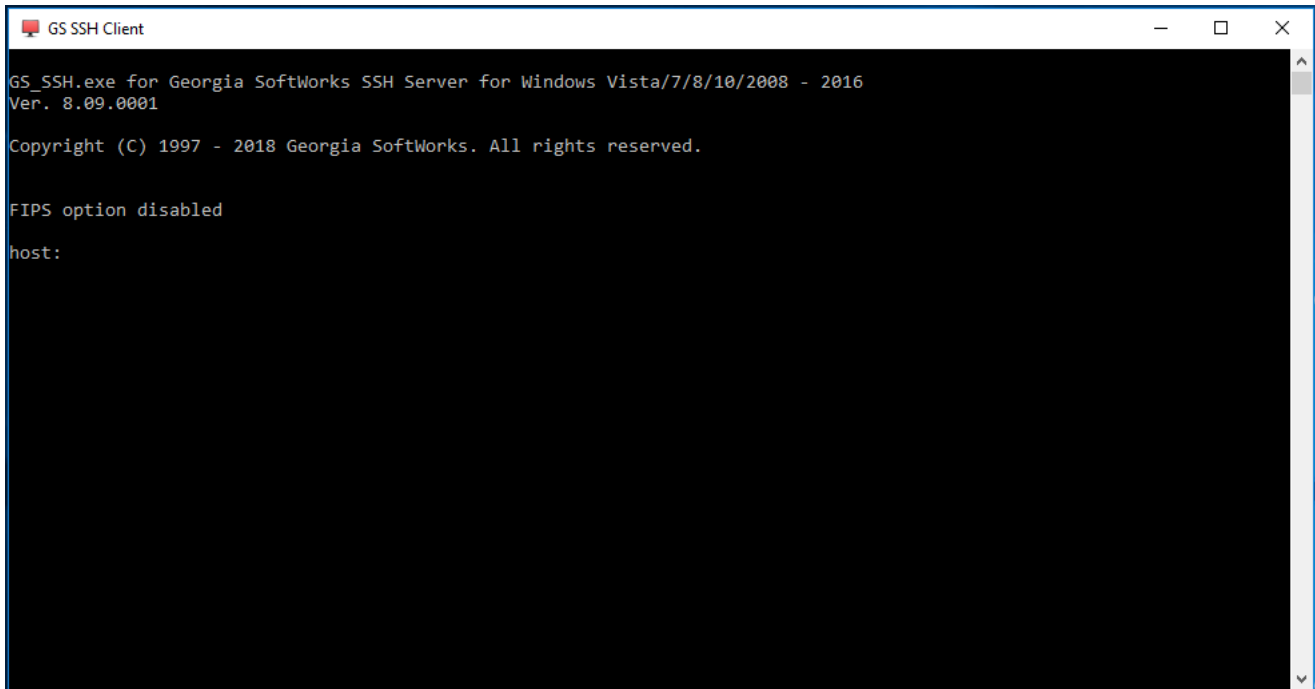
Table 7: GSW SSH Client Platforms

Please see the Georgia SoftWorks UTS User Guide for detailed description of client features and options.

## GSW DESKTOP CLIENT

In general, the GSW client installation procedures and features described in the GSW UTS User Manual are applicable to the GSW SSH Clients.

To invoke the GSW SSH Client, use the GS SSH Client shortcut in the GSW UTS program group. When connecting with the GSW SSH desktop client, you will get a logon banner similar to the one displayed below. The Host, Username, Password, and domain prompts are presented.

A screenshot of a Windows command prompt window titled "GS SSH Client". The window has a black background with white text. The text displayed is: "GS\_SSH.exe for Georgia SoftWorks SSH Server for Windows Vista/7/8/10/2008 - 2016", "Ver. 8.09.0001", "Copyright (C) 1997 - 2018 Georgia SoftWorks. All rights reserved.", "FIPS option disabled", and "host:". The window includes standard Windows window controls (minimize, maximize, close) in the top right corner and a vertical scrollbar on the right side.

```
GS_SSH.exe for Georgia SoftWorks SSH Server for Windows Vista/7/8/10/2008 - 2016
Ver. 8.09.0001

Copyright (C) 1997 - 2018 Georgia SoftWorks. All rights reserved.

FIPS option disabled

host:
```

Figure 54: GSW SSH Desktop Client

### Windows Mobile Clients

GSW provides SSH clients for Pocket PC/Windows Mobile class devices. Installation is as described in the GSW UTS User Manual. Items specific to the GSW SSH Pocket PC clients are noted below.

#### Windows Mobile

Upon installation of the GSW Windows Mobile client, you have the connection configuration similar as pictured below. The main item of interest is the Port selected to use for the SSH connection. The normal port used for SSH connections is port 22. Please configure as identified.

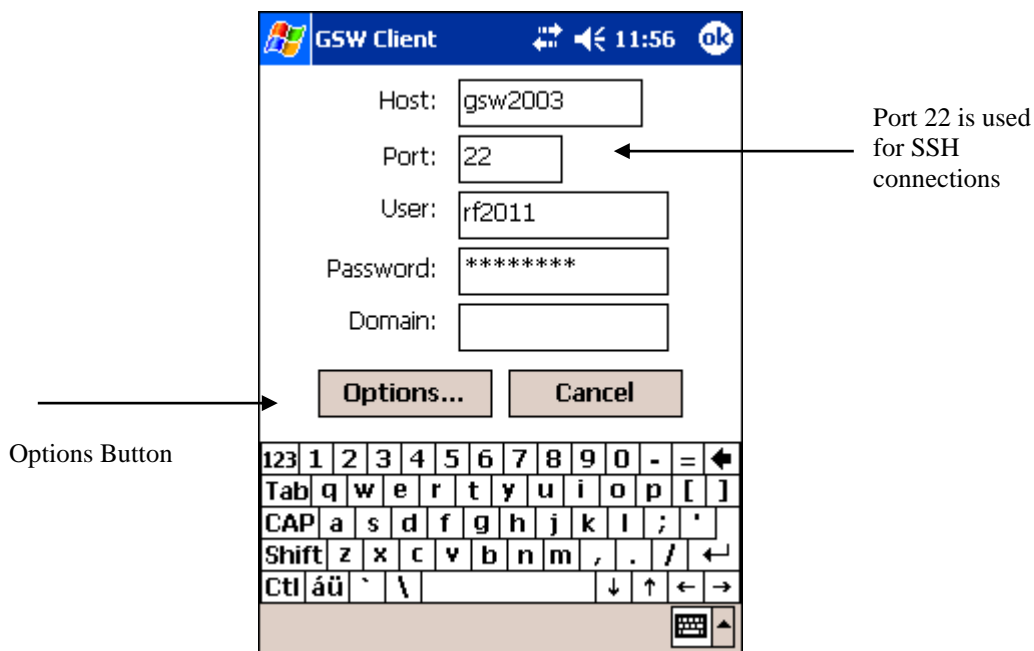


Figure 55: GSW PPC 2003 Client

To enable SSH encryption click on the Options button.



After clicking on the Options Button the following screen is displayed. The encryption combo box allows the options No encryption, 40-bit, 128-bit, SSH and FIPS SSH. Options selected that do not fit into the context of the GSW Server will result in a failed connection. For example, selecting FIPS SSH encryption when the GSW SSH server does not have FIPS enabled.

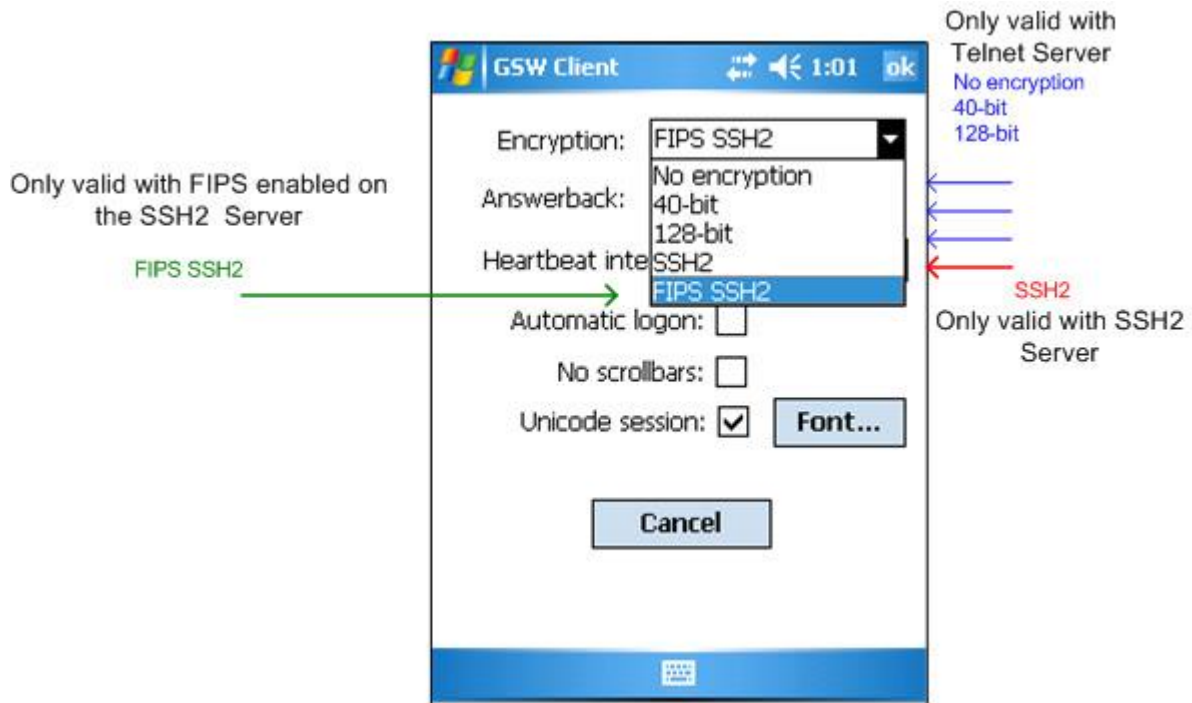
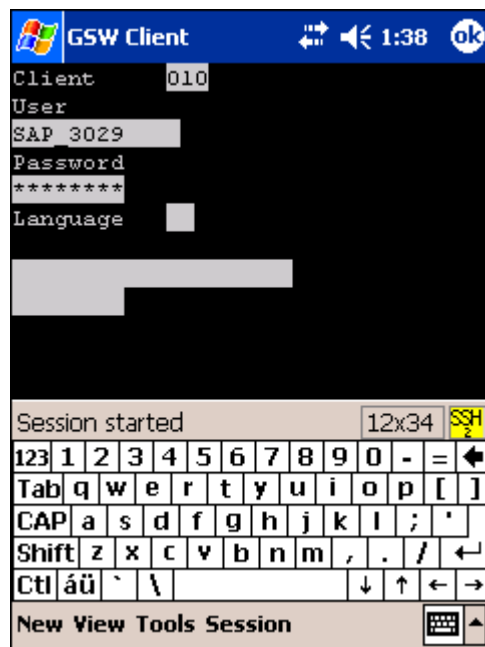


Figure 56: GSW PPC 2003 Client – Options

This is a screen shot of a PPC2003 connection to SAP via SAPConsole.



Note: The Yellow SSH symbol confirms that the SSH protocol is in use.

Figure 57: GSW PPC 2003 Client - SAPConsole - SSH

## Windows CE 4.2+ Devices

Georgia SoftWorks provides a Windows CE .NET 4.2+ SSH client. Below are some screen images of the GSW SSH Client in action on a Psion-Teklogix device.

Upon launching from the shortcut on the device desktop the initial screen (Figure 58) is displayed. From the Initial Screen you have the menu options File, View, Session and Help.

The Session menu (Figure 59) item provides the mechanism to Connect, Disconnect and to configure your session configuration settings.



Figure 58: Psion-Teklogix Initial Screen



Figure 59: Psion-Teklogix – Session Menu Items

By selecting the Session -> Settings the screen below (Figure 60) is presented allowing configuration of the Host, Port, User, Password and Domain. Selecting the Options button provides similar options as presented in the GSW Windows Mobile client (Figure 56).

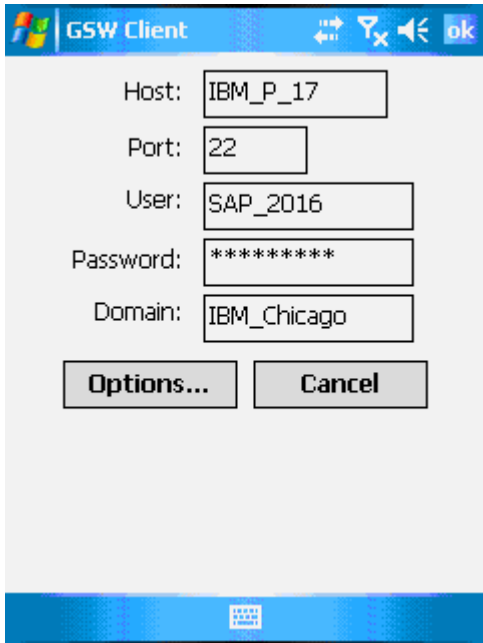


Figure 60: Psion-Teklogix Connection Settings

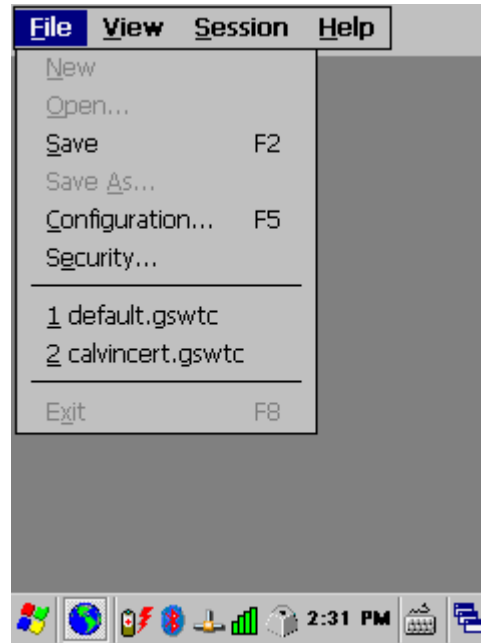


Figure 61: Psion-Teklogix – Save Settings

When the configuration is complete you can save the session configuration information by using the File menu item (Figure 61). You may recall the configuration and minimize the amount of data typed to connect. It also provides the flexibility to save several profiles if needed.

Using the Menu item Session->Connect, the connection is established and Figure 62 is an example of a connection to SAP via SAPConsole.

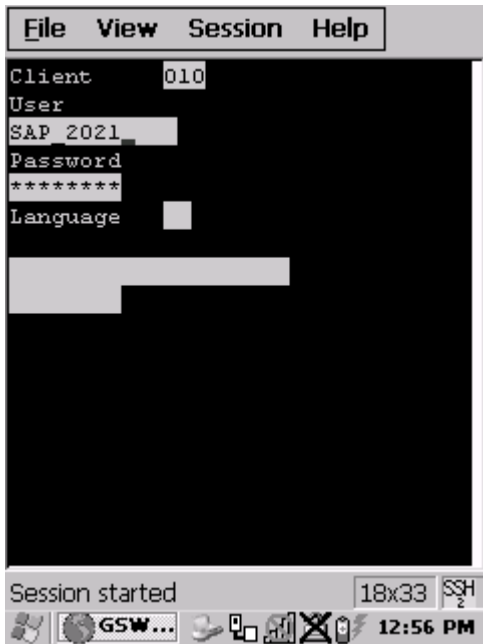


Figure 62: Psion-Teklogix running SAP via SAPConsole

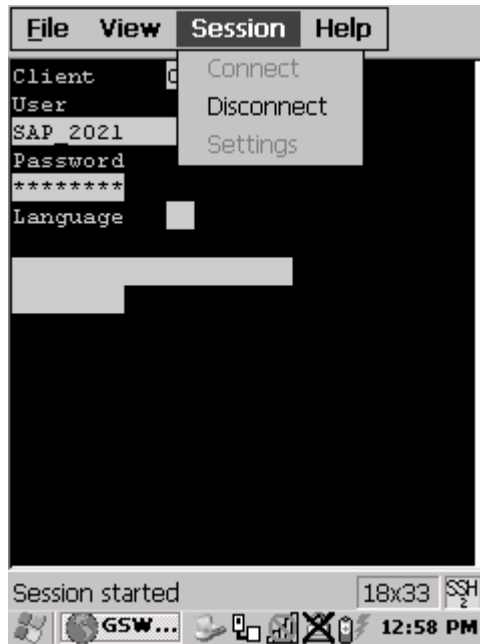


Figure 63: Psion-Teklogix Save Client Settings Menu

After the work is complete the session is disconnected by using the Menu item Session->Disconnect.

### Third Party SSH Clients

The GSW SSH Server allows connections from 3<sup>rd</sup> Party SSH Clients.

Please see the User’s Manual of the 3<sup>rd</sup> party SSH client of interest for operations of that client. We have included screen shots from three popular SSH clients operating with the GSW SSH Server.

Below is a screen shot of the SecureCRT SSH Client connected to the GSW SSH Server and running SAP via SAPConsole.

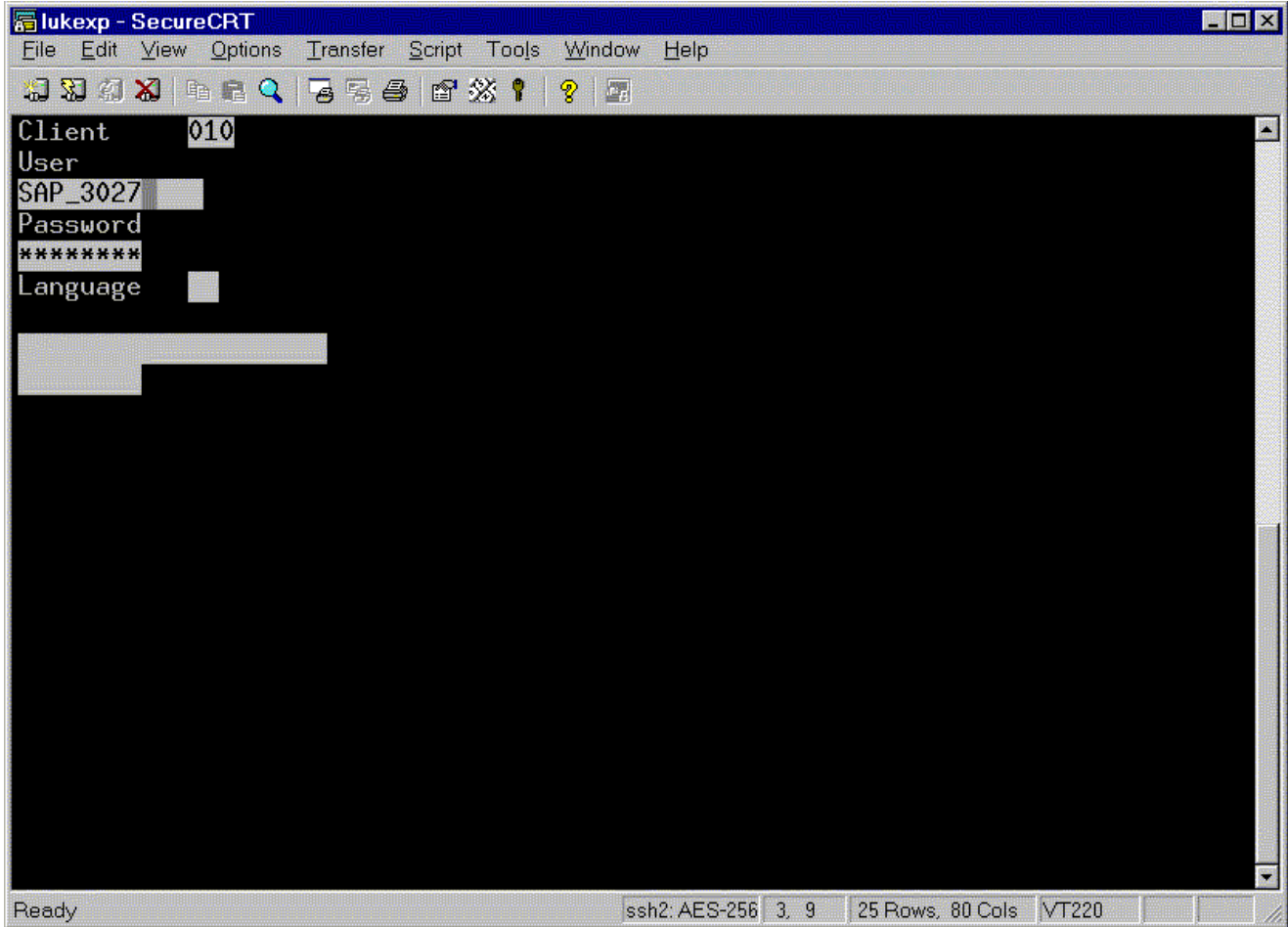
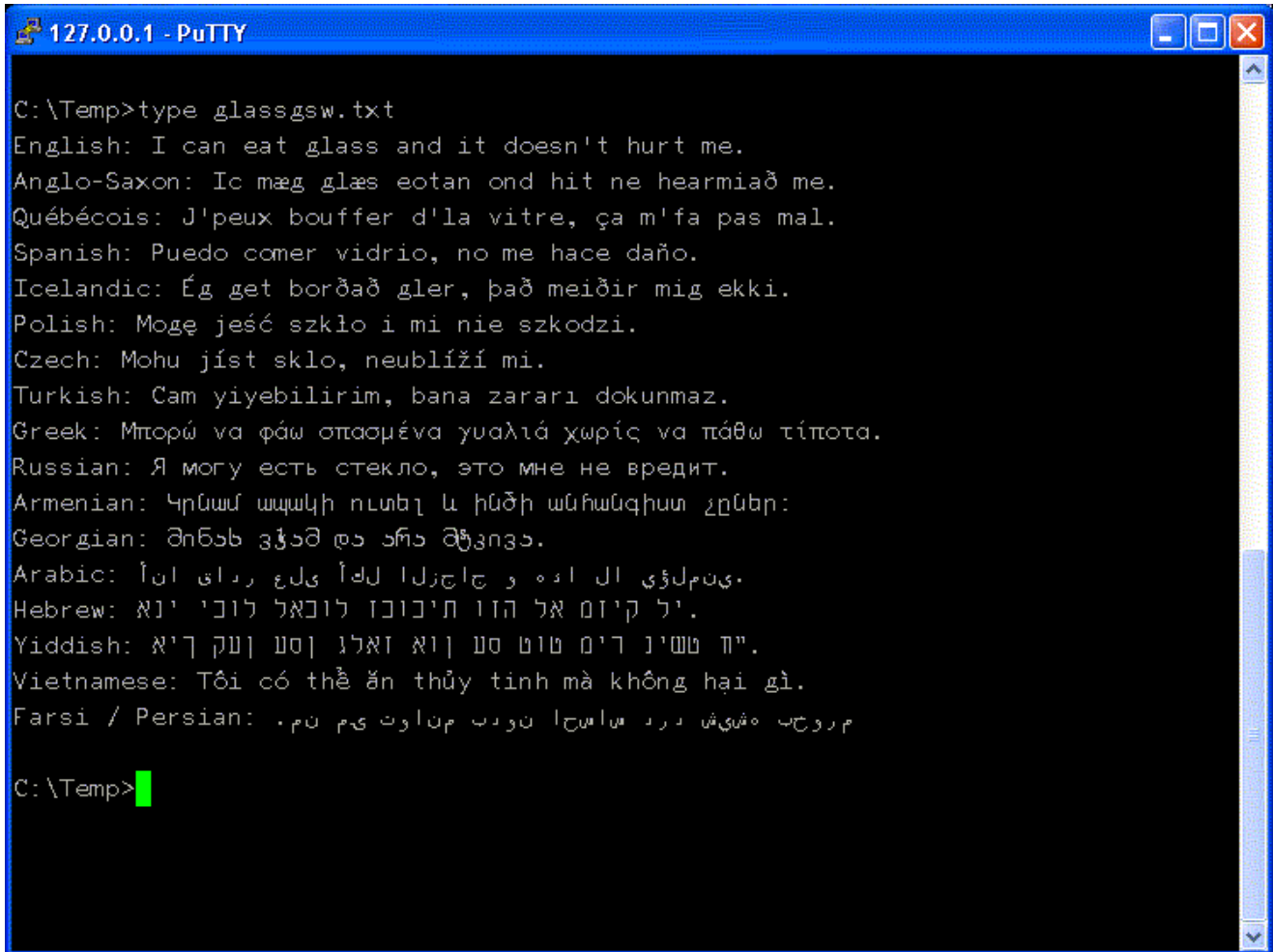


Figure 64: 3rd Party Client – SecureCRT – SAPConsole

Below is a screen shot of the PuTTY SSH Client displaying some of the GSW International character support.



```
C:\Temp>type glassgsw.txt
English: I can eat glass and it doesn't hurt me.
Anglo-Saxon: Ic mæg glæs eotan ond hit ne hearmiað me.
Québécois: J'peux bouffer d'la vitre, ça m'fa pas mal.
Spanish: Puedo comer vidrio, no me hace daño.
Icelandic: Ég get borðað gler, það meiðir mig ekki.
Polish: Mogę jeść szkło i mi nie szkodzi.
Czech: Mohu jíst sklo, neublíží mi.
Turkish: Cam yiyebilirim, bana zararı dokunmaz.
Greek: Μπορώ να φάω οπασμένα γυαλιά χωρίς να πάθω τίποτα.
Russian: Я могу есть стекло, это мне не вредит.
Armenian: Կրճա՛մ ապակի ուտել և ինձի ա՛նհաճախու չընել:
Georgian: მინას ვჭამ და არა მტუკნდა.
Arabic: يمكنني أكل زجاج ولا يؤذي.
Hebrew: יל ק'יזם אל הזו ת'יכוננו לונאל לונבי י'א.
Yiddish: ײן טשײן ד'ם טוט עס און זאלן עס אן ק'אן.
Vietnamese: Tôi có thể ăn thủy tinh mà không hại gì.
Farsi / Persian: من می‌توانم شیشه را بخورم و به من آسیبی نمی‌رسد.
```

Figure 65: 3rd Party Client - PuTTY - Unicode

Below is a screen shot of the F-Secure SSH Client connected to the GSW SSH Server and running SAP via SAPConsole.

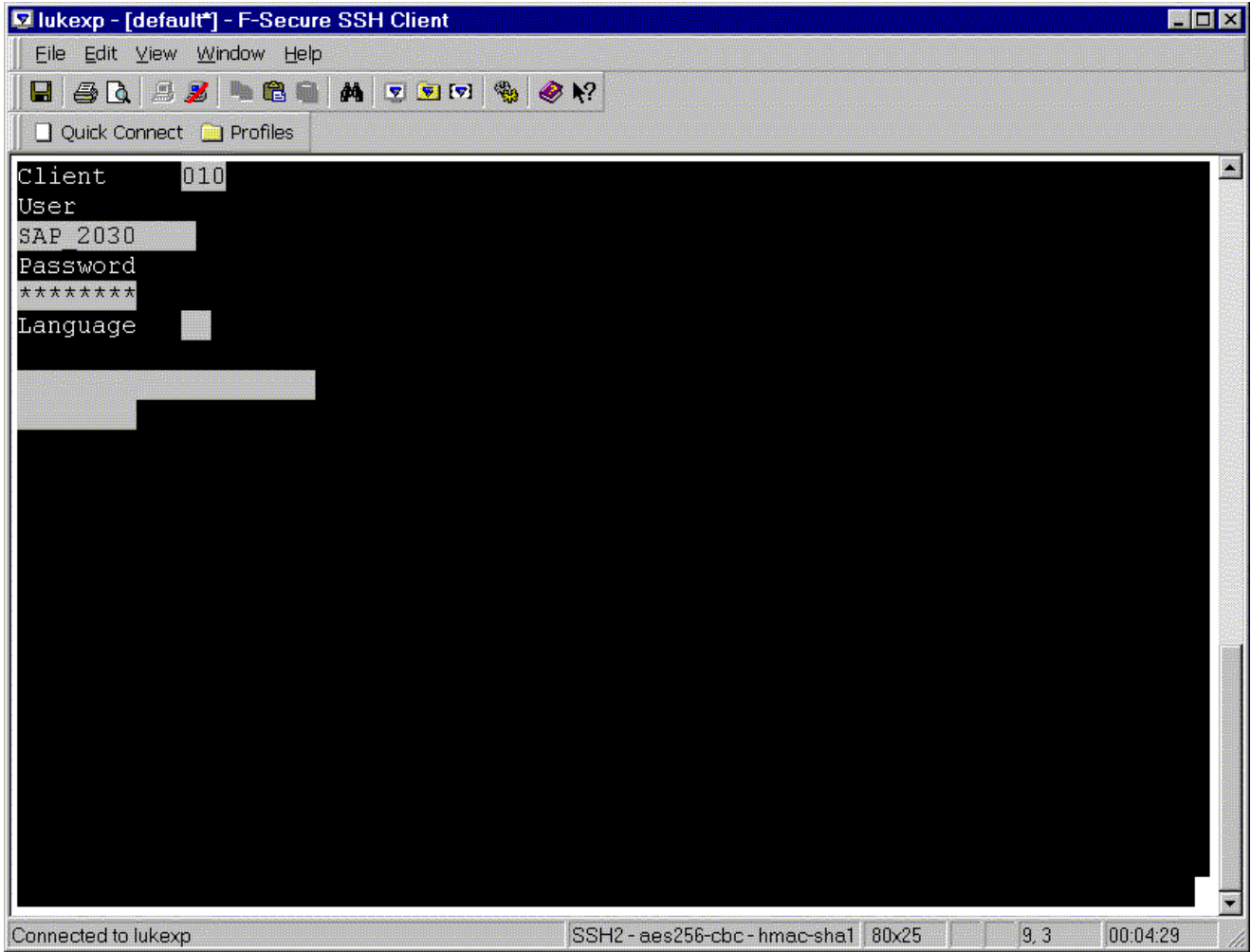


Figure 66: 3rd Party Client - F-Secure SSH Client

### **Specify Domain with a 3<sup>rd</sup> Party Client**

A user account's domain can be specified in the SSH client's user name field. If a domain is not specified then the GSW UTS will use the default domain configured in the UTS registry. If a UTS default domain is not configured and a domain is not specified in the SSH client's user name field then the system will attempt to validate the user account logon using the local account database.

Use the following syntax to specify the domain in the SSH client's user name field:

```
username@domainname
```

Where `username` is the name of the user and `domainname` is the name of the domain.

If a default domain is specified in the UTS registry then the domain entered above will take precedence. Please see the GSW UTS User Manual for more information.



## Registry Variables

Many registry variables exist for provisioning the system. Registry variables are an excellent method to configure software while utilizing skills already learned by the system administrator. There is no need to learn yet another interface to provision the software. Here is a list of the registry variables and a brief description of their use. Please see the appropriate section in this User Manual for complete descriptions.

All Registry values used by the Georgia SoftWorks SSH Server are stored in the following Registry path.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Georgia SoftWorks\GSW_SSHD\Parameters
```

- `bAllowServiceSCP` - Allow/Disallow user's use of the "SCP" channel. Default=0. (Page 42)
- `bAllowServiceShell` - Controls use of the "Shell" channel as defined. Default=1. (Page 43)
- `bAllowSFTP` - Control use of the SFTP subsystem. Default=0. (Page 44)
- `bRestrictedSFTP` - Disables username and domain restrictions. Default=1 (Page 46)
- `bDisableFIPS` - This parameter must be set to 1 to use algorithms which are not on the FIPS list. The default value of this parameter is 0.
- `szSFTPRoot` - Specifies the local folder path to use with SFTP. Default=empty string. (Page 45)
- `bEnableLogonCertificate` - Enables/Disables use of Logon Certificates. Default=1. (Page 48)
- `bEnableLogonPublicKey` - Enables/Disables use of Public Key authentication. Default=1. (Page 49)
- `bEnableLogonPassword` - Enables/Disables use of User Name/Password authentication. Default=1. (Page 49)
- `bEnableLogonGSSAPI` - Enables/Disables use of GSSAPI for logon. Default=1. (Page 50)
- `szBindIPv4` - Specifies address to listen on for IPv4 connections. Default=empty string (Page 51)
- `szBindIPv6` - Specifies address to listen on for IPv6 connections. Default=empty string (Page 51)
- `bAllowServiceExecute` - Allows/Disallows user's use of the "exec" channel. Default=0 (Page 52)
- `bAllowRSAHostKey` - Enables/Disables use of RSA Host Key. Default=1 (Page 53)

- `bAllowDSAPublicKey` - Enables/Disables use of DSA Host Key Default=1 (Page 54)
- `bAllowECDSAPublicKey` - Enables/Disables use of ECDSA Host Keys Default=1 (Page 54)
- `szCiphers`- Specify the list of Ciphers the server can offer the client.

**Default=** " aes256-gcm@openssh.com, chacha20-poly1305@openssh.com, aes256-ctr, aes192-ctr, 3des-cbc, aes128-ctr, aes128-gcm@openssh.com, aes256-cbc, rijndael256-cbc, rijndael-cbc@lysator.liu.se, aes192-cbc, rijndael192-cbc, aes128-cbc, rijndael128-cbc, cast128-cbc, blowfish-cbc " (Page 56)

- `szKexAlgorithms` - Specify the list of Key Exchange Algorithms the server can offer the client

**Default=** " curve25519-sha256, curve25519-sha256@libssh.org, diffie-hellman-group18-sha512, diffie-hellman-group16-sha512, diffie-hellman-group-exchange-sha256, diffie-hellman-group14-sha256, ecdh-sha2-nistp521, ecdh-sha2-nistp384, ecdh-sha2-nistp256, diffie-hellman-group-exchange-sha1, diffie-hellman-group14-sha1, diffie-hellman-group1-sha1" (Page 57)

- `szMacs` - Specify the list of Message Authentication Code algorithms the server can offer to the client.

**Default=** " hmac-sha2-512-etm@openssh.com, hmac-sha2-256-etm@openssh.com, hmac-sha2-512, hmac-sha2-256, hmac-sha1-etm@openssh.com, hmac-sha1, hmac-sha1-96, hmac-md5, none" (Page 58)

- `bEnableWODLog` - Turn Logging ON for SSH internals activity. Default = 0 (Page 68 )
- `dwInactivityTimeout` - Reserved, do not change
- `dwLogonTimeout` - How long SSH Server waits at logon prompt before dropping connection. Default 120.
- `szServerAddress` - Reserved, do not change
- `szServerDSAKeyFile` - Location of SSH Servers DSA private key file in PEM format. The file is encrypted. (Page 67)
- `szServerECDSAKeyFile` - Location of SSH Servers ECDSA private key file in PEM format. The file is encrypted. (Page 68)
- `szServerRSAKeyFile` - Location of SSH Servers RSA private key file in PEM format. The file is encrypted. (Page 66)
- `szWODLogFile` - Path and File Name of the SSH internal activity log file. To enable the log `bEnableWODLog` must be set to 1. (Page 71)

`usGSWSSHDPort` - The port number clients will be connecting to. (Page 42)  
Default = 22(decimal) is the standard port assigned to SSH.

# SSH Algorithms Support

## Key Exchange Algorithms

Key exchange algorithms are used to exchange cryptographic keys between the SSH Server and the SSH client.

<b>GREEN</b> – Strong Enough to be considered SAFE <b>Purple</b> – Questioned by 3 <sup>rd</sup> parties <b>Black</b> – not researched yet. <b>RED</b> – Should not be used	G S W S S H S e r v e r		G S W C l i e n t s V 8 . 1 0 . 0 0 0 3					
	8.10.0003		Desktop		CE 4.2	CE 5.0+		G S W C o n n e c t B o t
		FIPS		FIPS	FIPS		FIPS	V1.9.9
<b>Host Key Algorithms</b>								
<b>ssh-ed25519</b>								✓
<b>rsa-sha2-512</b>	✓	✓	✓	✓		✓	✓	✓
<b>rsa-sha2-256</b>	✓	✓	✓	✓		✓	✓	✓
ecdsa-sha2-nistp521	✓	✓	✓	✓				✓
ecdsa-sha2-nistp384			✓	✓				✓
ecdsa-sha2-nistp256			✓	✓				✓
<i>ssh-rsa</i>	✓	✓	✓	✓	✓	✓	✓	✓
<i>ssh-dss</i>	✓	✓	✓	✓	✓	✓	✓	✓
<b>Key exchange algorithms</b>								
<b>curve25519-sha256@libssh.org</b>	✓	✓	✓	✓	✓	✓	✓	✓
<b>curve25519-sha256</b>	✓	✓	✓	✓	✓	✓	✓	✓
<i>diffie-hellman-group-exchange-sha256</i>	✓	✓	✓	✓		✓	✓	✓
<b>diffie-hellman-group14-sha256</b>	✓	✓	✓	✓	✓	✓	✓	
<b>diffie-hellman-group16-sha512</b>	✓	✓	✓	✓		✓	✓	✓
<b>diffie-hellman-group18-sha512</b>	✓	✓	✓	✓		✓	✓	✓
ecdh-sha2-nistp521	✓	✓	✓	✓				✓
ecdh-sha2-nistp384	✓	✓	✓	✓				✓
ecdh-sha2-nistp256	✓	✓	✓	✓				✓
ext-info-c	✓	✓	✓	✓	✓	✓	✓	✓
<i>diffie-hellman-group-exchange-sha1</i>	✓	✓		✓	✓	✓	✓	✓
<i>diffie-hellman-group-14-sha1</i>	✓	✓		✓	✓	✓	✓	✓
<i>diffie-hellman-group1-sha1</i>	✓	✓		✓	✓	✓	✓	✓

Table 8: SSH Host Key and Key Exchange Algorithms

# Ciphers

Ciphers are algorithms used for performing encryption or decryption.

Ciphers	G S W S S H S E R V E R		G S W C l i e n t s V 8 . 1 0 . 0 0 0 3						
	8.10.0003		Desktop		CE 4.2		CE 5.0+		G S W C o n n e c t B o t V 1 . 9 . 9
		FIPS		FIPS		FIPS		FIPS	
<b>chacha20-poly1305@openssh.com</b>	✓		✓		✓		✓		
<b>aes256-gcm@openssh.com</b>	✓	✓	✓	✓					
<b>aes128-gcm@openssh.com</b>	✓	✓	✓	✓					
<b>aes256-ctr<sup>11</sup></b>	✓	✓	✓	✓			✓	✓	✓
<b>aes128-ctr<sup>12</sup></b>	✓	✓	✓	✓			✓	✓	✓
<b>aes192-ctr<sup>13</sup></b>	✓	✓	✓	✓			✓	✓	✓
<b>aes256-cbc</b>	✓	✓	✓	✓			✓	✓	✓
<b>aes192-cbc</b>	✓	✓	✓	✓			✓	✓	✓
<b>aes128-cbc</b>	✓	✓	✓	✓			✓	✓	✓
<b>3des-ctr<sup>14</sup></b>									✓
<b>3des-cbc</b>	✓	✓	✓	✓	✓		✓	✓	✓
<b>blowfish-ctr</b>									✓
<b>blowfish-cbc</b>	✓		✓						✓
<b>rijndael256-cbc</b>	✓	✓	✓	✓			✓	✓	
<b>rijndael192-cbc</b>	✓	✓	✓	✓			✓	✓	
<b>Rijndael128-cbc</b>	✓	✓	✓	✓			✓	✓	
<b>rijndael-cbc@lysator.liu.se</b>	✓	✓	✓	✓					
<b>cast128-cbc</b>	✓		✓						

Table 9: SSH Ciphers

<sup>11</sup> aes256-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>12</sup> aes128-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>13</sup> aes192-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

<sup>14</sup> 3des-ctr is safe when used with hmac-sha2-256-etm@openssh.com or hmac-sha2-512-etm@openssh.com

## HMACs - Hash Message Authentication Code

A Hash Message Authentication Code is method for message authentication using cryptographic hash functions combined with a secret key that is shared.

<b>GREEN</b> – Strong Enough to be considered SAFE <b>Purple</b> – Questioned by 3 <sup>rd</sup> parties <b>Black</b> – not researched yet. <b>RED</b> – Should not be used	G S W S S H S E R V E R		G S W C L I E N T S						
	S e v e r		V 8 . 1 0 . 0 0 0 3						
	8 . 1 0 . 0 0 0 3		D e s k t o p		C E 4 . 2		C E 5 . 0 +		G S W C o n n e c t B o t
MACs	F I P S	F I P S	F I P S	F I P S	F I P S	F I P S	F I P S	V 1 . 9 . 9	
hmac-sha2-512-etm@openssh.com	✓	✓	✓	✓			✓	✓	✓
hmac-sha2-256-etm@openssh.com	✓	✓	✓	✓			✓	✓	✓
hmac-sha2-512	✓	✓	✓	✓			✓	✓	✓
hmac-sha2-256	✓	✓	✓	✓			✓	✓	✓
hmac-sha1	✓	✓	✓	✓	✓		✓	✓	✓
hmac-sha1-96	✓	✓	✓	✓					✓
hmac-md5	✓		✓		✓		✓		✓
hmac-md5-96									✓
None	✓		✓		✓		✓		
hmac-sha1-etm@openssh.com	✓	✓	✓	✓	✓		✓	✓	✓

Table 10: SSH HMACs

## Host Key Types

The purpose of a Host Key is to ensure that when you connect to a remote host, it is actually the host you want to connect. It is the SSH Server’s public key and is used by the SSH client to decrypt the authentication message sent from the server when establishing a connection.

The public key / certificate formats supported by the GSW SSH Server are shown below.

<b>GREEN</b> – Strong Enough to be considered SAFE <b>Purple</b> – Questioned by 3 <sup>rd</sup> parties <b>Black</b> – not researched yet. <b>RED</b> – Should not be used	G S W S S H S E R V E R		G S W C L I E N T S						
	S e v e r		V 8 . 1 0 . 0 0 0 3						
	8 . 1 0 . 0 0 0 3		D e s k t o p		C E 4 . 2		C E 5 . 0 +		G S W C o n n e c t B o t
Public Key Algorithms	F I P S	F I P S	F I P S	F I P S	F I P S	F I P S	F I P S	V 1 . 9 . 9	
rsa-sha2-512	✓	✓	✓	✓			✓	✓	✓
rsa-sha2-256	✓	✓	✓	✓			✓	✓	✓
ssh-ed-25519									✓
ssh-rsa	✓	✓	✓	✓	✓		✓	✓	✓
ssh-dss	✓	✓	✓	✓	✓		✓	✓	✓

Table 11: SSH Public Key Algorithms

## Compression

<b>GREEN</b> – Strong Enough to be considered SAFE <b>Purple</b> – Questioned by 3 <sup>rd</sup> parties <b>Black</b> – not researched yet. <b>RED</b> – Should not be used	G S W S S H S e r v e r		G S W C l i e n t s V 8 . 1 0 . 0 0 0 3						
	8.10.0003		Desktop		CE 4.2		CE 5.0+		G S W C o n n e c t B o t
		FIPS		FIPS		FIPS		FIPS	V1.9.9
<b>Compression</b>									
<b>None</b>	✓	✓	✓	✓				✓	
<b>zlib</b>	✓	✓	✓	✓				✓	

Table 12: SSH Compression

## FIPS 140-2 Resources

Additional information about FIPS and NIST can be found using the following links.

<http://csrc.nist.gov/publications/PubsFIPS.html>

Certificate numbers

Certificate Numbers	Descriptions
<b>#560</b>	Certificate #560 Windows CE and Windows Mobile Enhanced Cryptographic Provider (RSAENH) (Software Versions: 5.01.01603 [1], 5.00.911762 [1], 5.04.17228 [2] and 5.05.19202 [2]) <a href="http://csrc.nist.gov/publications/PubsFIPS.html">http://csrc.nist.gov/publications/PubsFIPS.html</a>
<b>#825</b>	Certificate #825 Windows CE and Windows Mobile Enhanced Cryptographic Provider (RSAENH) (Software Version: 6.00.1937) <a href="http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140crt/140crt825.pdf">http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140crt/140crt825.pdf</a>
<b>#918</b>	Certificate #918 OpenSSL FIPS Object Module) <a href="http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140crt/140crt918.pdf">http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/140crt/140crt918.pdf</a>

Table 13: FIPS 140-2 certificate links

## G S W S S H S e r v e r S u b s c r i p t i o n

The GSW Subscription plan provides access to the most current versions of the software as well as priority support.

In general, Georgia SoftWorks releases a new version as soon as new features are ready rather than waiting for quarterly or annual releases. Due to our development and release generation methods and JIT User Manual production, we can release software on a much more frequent basis than other organizations. As soon as features or defect resolutions are Alpha and Beta tested we generate a release. This provides our customers with features much quicker than the “grouping” or “scheduling” method used by other companies.

The GSW SSH Server (and Rocket Pack, RF DTIO) Subscription grants access to free version upgrades for the duration of the subscription. The duration is either 1, 2 or 3 years. This is good as you can obtain new versions of the software at your convenience, obtaining all new features and defect resolutions.

**NOTE:** New versions can be downloaded from our web site at you convenience.

The GSW Subscription plan is an excellent value. Even if you upgrade the software once every few years you will save with the subscription.

Version Upgrade Pricing <b>with</b> Subscription Plan	
TIME FROM DATE OF PURCHASE	PRICE
For the Duration of Plan (1, 2 and 3 year plans are available).	<b>Free</b>

Table 14: Version Upgrade Pricing **with** GSW Subscription Plan

The pricing for version upgrades without the Subscription is based on the period of time since the date of the original purchase or last version upgrade.

Version Upgrade Pricing <b>without</b> Subscription Plan	
TIME FROM DATE OF PURCHASE	PRICE
Less than 60 days	Free
Greater than 60 days but less than 1 year	50% of the current list
Greater than 1 year	90% of the current list

Table 15: Version Upgrade Pricing **Without** Subscription Plan



## HOW TO UPDATE THE SOFTWARE

1. Download the software.
2. Make sure the SSH Server is not in use.
3. Run the Setup Program for the Update as done in the original installation.
4. You may specify the same or different installation folder.

## HOW TO RENEW THE GSW Subscription

Please use the following procedure when renewing the GSW SSH Server or Rocket Pack Subscription.

Step	Who	Action
1.	GSW	Send notice to customer indicating that the subscription is about to expire. The notice is sent approximately 4 to 8 weeks prior to the expiration of the plan.
2.	Customer	Places order for new subscription
3.	GSW	Confirms Order
4.	GSW	Ships current software, documentation and new Floating License (if applicable)
5.	Customer	Install new Floating License (and software if desired)
6.	Customer	Ships OLD Floating License back to GSW

Table 16: Steps to Renew the GSW Subscription Plan

## SSH Server Folder Layout

The Installation folder of the GSW UTS is as follows

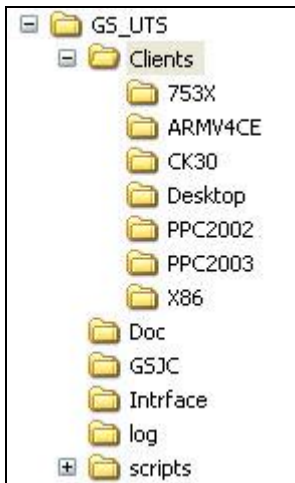


Figure 67: Installation Folder Layout of the GSW UTS

The folders of interest are:

- Clients: Contains all the GSW clients for the SSH Server and the Telnet Server. These files are needed for automatic update of our client software.
- 753x Contains the GSW Client for Teklogix 753x devices.
- ARMV4CE Contains the GSW Client for ARM devices
- CK30 Contains the GSW Client for Intermec CK30 devices
- Desktop Contains the GSW clients that run on Windows Desktops.
- PPC2002 GSW Clients for Windows Pocket PC 2002 class devices
- PPC2003 GSW Clients for Windows Pocket PC 2003 class devices.
- X86 Contains the GSW Client for x86 based devices
- Doc: Contains the documentation for your viewing or printing.
- GSJC Contain the files for the GS Java Client and Applet
- Log Contains the GSW UTS Log files to provide to the GSW Technical Support Group in the event of a problem. See page 116 for more information.
- Scripts This is where your logon scripts will reside. See GSW UTS User Manual.

The installation folder layout of the GSW SSH Shield is as follows under the Windows\Program Files (x86) folder.



Figure 68: Installation Folder Layout of the GSW SSH Shield

The Georgia SoftWorks UTS logs folder contains the GSW SSH Server log files to provide to the GSW Technical Support Group in the event of a technical problem.

## System Signature - IMPORTANT PLEASE READ

NOTE: This section only applies to Software Registration

The registration software obtains a system signature that is unique to your system. This signature is an added security measure to inhibit unauthorized personnel to obtain working copies of the GSW SSH Server.

The signature is comprised of hardware and software identifiers that exist on your system that make the target system unique. These identifiers are hashed into a Product ID and a Serial Number can be generated from this Product id.

If major hardware components of your system are removed, replaced or modified your **Serial Number** may discontinue to work and you may need a new **Serial Number** to obtain access to the SSH Server. Please contact Georgia SoftWorks Technical Support if needed.

## Technical Support

In order to keep Technical Support **Free** please help keep our cost down.

- Gather all relevant system and environment information.
- Write your question down. This not only helps us but also helps you in articulating the question.

### Provide Log Files To GSW Technical Support

A typical sequence when GSW Technical Support needs the logs files are to delete the log files, reproduce the behavior in question and email the log files, which are recreated during the test, to GSW Support.

### Email Support Tips:

To expedite support for suspected problems please perform the following test steps below to help us diagnose the issue.

1. Disconnect all users. Make sure that no other user connects at the time of the test.
2. Wait 5 minutes
3. Delete the Log files

Delete all log files from the GSW UTS Server installation 'Log' subdirectory on the computer running the GSW Universal Terminal Server. (Usually `c:\GS_UTS\Log`)

4. To expedite resolution, reboot the Server if possible
5. Duplicate the problem.
6. The log files are automatically re-created. Send us the files using the [GSW Ticket System](#)
7. Please also include
  - a. A description of the problem including User ID's, Domain and IP Addresses
  - b. The logon script associated with the user experiencing the problem. (That is the `c_start.bat` or the `k_start.bat` file that resides in the scripts folder in the GSW UTS directory
  - c. And of course your contact information.

Again, send us the files using the [GSW Ticket System](#). We try to respond within 24 hours.

Or Call **706.265.1018 EST, M-F 9:00 a.m. to 5:00 p.m. and have your Product ID ready**