

System iNtrusion Analysis & Reporting Environment

SNARE Generator User Manual

INTER*S***ECT**
ALLIANCE

Documentation History

Version No.	Date	Edits	By whom
0.9	15 March 2004	First draft for the Snare Generator - User Manual.	George Cora
1.0	18 March 2004	Final release version.	Leigh Purdie
2.0	2 April 2005	Minor Rewording.	infofocus.com
2.1	16 September 2005	Updates for new version (1.2)	George Cora
2.2	30 November 2005	Formatting changes	George Cora
2.3	26 September 2006	Updates for new version (1.5)	George Cora
2.4	16 March 2007	Updates for new version (2.0)	George Cora
2.5	21 September 2007	Two small spelling fixes	Leigh Purdie

© 1999-2007 Intersect Alliance Pty Ltd. All rights reserved worldwide.

Intersect Alliance Pty Ltd shall not be liable for errors contained herein or for direct, or indirect damages in connection with the use of this material. No part of this work may be reproduced or transmitted in any form or by any means except as expressly permitted by Intersect Alliance Pty Ltd. This does not include those documents and software developed under the terms of the open source General Public Licence, which covers the Snare agents and some other software.

The Intersect Alliance logo and Snare logo are registered trademarks of Intersect Alliance Pty Ltd. Other trademarks and trade names are marks' and names of their owners as may or may not be indicated. All trademarks are the property of their respective owners and are used here in an editorial context without intent of infringement. Specifications and content are subject to change without notice.

About this guide

This guide provides you with step-by-step instructions on how to install, configure and use the SNARE Generator. The development of the 'SNARE Generator' allows for logs to be artificially generated and sent either via TCP or UDP to a remote collection host. The Snare Generator will run on Linux only, and will simulate Windows, Solaris, IRIX, CISCO PIX, CISCO Router, Cyberguard, Universal Log, SNORT, IIS Web Log, IP Tables and SYSLOG events.

Other guides that may be useful to read include:

- SNARE Server User's Guide.
- Installation Guide to the SNARE Server.
- SNARE Server Troubleshooting Guide.
- The SNARE Toolset - A White Paper.

Table of contents:

1 Introduction.....	4
2 Initial requirements.....	5
3 Using the SNARE generator.....	7
3.1 Configuration.....	7
3.2 Operation.....	9
4 SNARE Server.....	11
5 About Intersect Alliance.....	13
6 Appendix A - Event Log Formats.....	14

1 INTRODUCTION



The SNARE Generator has been designed to artificially generate events and send them over the network to a SNARE Server. As at version 2.0 of the SNARE Generator release, 11 log types are able to be artificially generated, namely: Windows, Solaris, IRIX, CISCO PIX, CISCO Router, Cyberguard, Universal Log, SNORT, IIS Web Log, IP Tables and SYSLOG events. This tool was originally designed to test the SNARE Server operation, but may be useful to parties who use other collection servers.

The artificially generated events will be sent over the network to UDP/TCP port 6161 or SYSLOG (UDP) port 514, the listening port for the SNARE Server. SYSLOG events such as PIX. IPTables events will be sent to the SYSLOG port, namely UDP port 514. These ports are not configurable.

The SNARE Generator has been developed using Glade-2 and gcc tools, for the Linux GTK+/Gnome environment.

2 INITIAL REQUIREMENTS



▶ WHAT YOU NEED... Please ensure that you have the following available:

- The latest SNARE Generator package, available from <http://www.intersectalliance.com/projects/index.html>
- A Linux system running GTK+ Vers 2.2 or greater. This application is designed for Linux only, at this time.
- Root-level access to the system.
- At least 2 Megabytes of free disk space on your system.

▶ HOW TO... Installation instructions follow:

In order to install the SNARE Generator, simply download the source tar file, and run the following commands (as root):

```
# tar zxvf snare-generator-2.0.tar.gz
# cd snaregen-2.0/
# make
# make install
# snaregen
```

In order to run the executable, simply type 'snaregen' as any normal user. Root or administrative level privileges are not required. A graphical user interface (GUI) similar to that shown below in Figure 1 will appear, once this command has successfully executed.

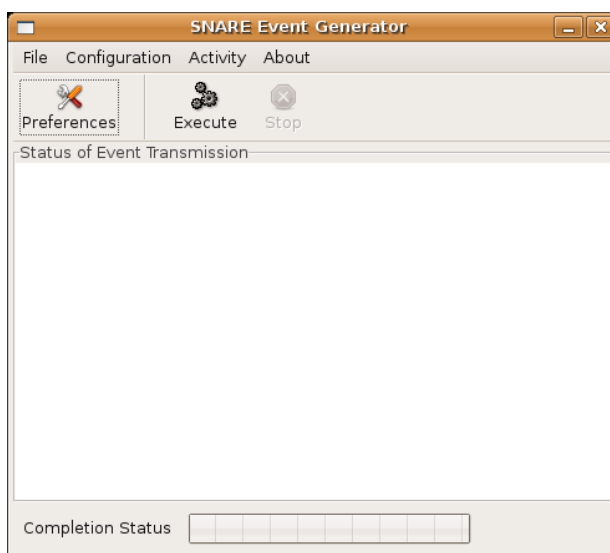


Figure 1 SNARE Generator Main Window

3 USING THE SNARE GENERATOR

3.1 CONFIGURATION

The SNARE Generator may be configured in a number of ways. Unless the preferences are set, the default values shown below will be used. In almost all cases, some changes will need to be made to the default configuration in order for the SNARE Generator to work. Figure 2 shows the configuration items available, and their default values.

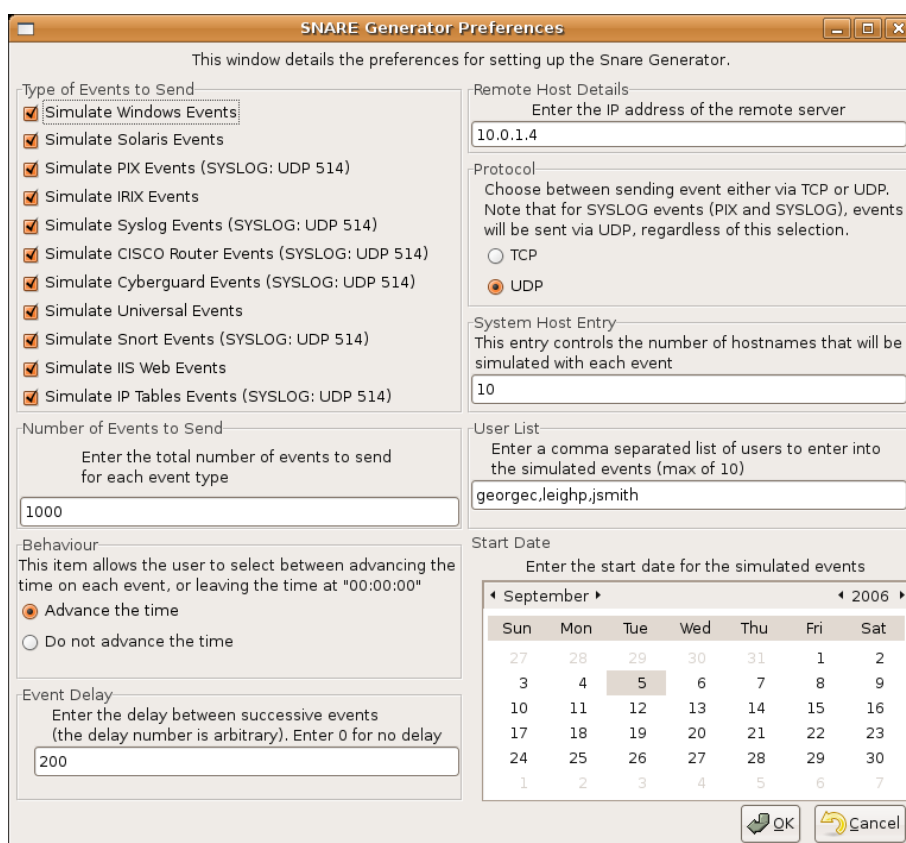


Figure 2 SNARE Generator Preferences

The configuration items available to the SNARE Generator user, either through the 'Configuration' menu item or the main toolbar include the following.

- a. **Types of Events to Send.** As at Version 2.0, there are 11 log types which may be generated, namely Windows, Solaris, IRIX, CISCO PIX, CISCO Router, Cyberguard, Universal Log, SNORT, IIS Web Log, IP Tables and SYSLOG events. Within these types of event logs, certain types of logs are generated and sent over the network. The 'behavior' item below further describes the event log types which are generated. Note that for the Windows, IRIX, Universal Log, IIS Web and Solaris event types, events are sent to the default SNARE Server port of 6161, either via UDP or TCP. For the SYSLOG events, these are sent to the default SYSLOG port of UDP 514. These port values are not configurable. The generation of certain types of events may be excluded by de-selecting the relevant check box as shown in Figure 2. *The default behavior is to turn on all 11 types of event generation.*

- b. **Number of Events to Send.** The raw number of logs that will be sent by the SNARE Generator can be controlled by this setting. If the number entered into this item is not greater than 0, then a warning will be displayed when the 'OK' button is clicked. A maximum of 32 digits may be entered in this item. *The default behavior is to generate 1000 events.*
- c. **User List.** When generating Windows, IRIX and Solaris events, users are included within the event records. The list defined in this entry item will be the users that get cycled through when generating the events. *The default behavior is to generate a list of the following users: georgec, leighp, jsmith.*
- d. **Behavior.** This item will either advance the date and time, or will leave the time at 00:00:00. If the "Advance the Time" button is selected, every few events, the time will be advanced 1 second from 00:00:00 from a starting date as per the "Start Date" setting. The date will also be advanced, once the time has reached 23:59:59. If the other button is selected the time will be left untouched at 00:00:00 and the date also untouched as per the "Start Date" entry below.
- e. **Remote Host Details.** This item details the remote host's (valid format) IP address or resolvable name. In the case of the IP address, any valid format IP address will suffice for it to be accepted by the SNARE Generator. If a DNS name is used, then it must be resolvable. If either of these conditions are not met, then an error will be displayed if the 'OK' button is selected. *The default behavior is to enter the IP address of 10.0.1.3, and in all cases this will most likely be different for each site.*
- f. **Start Date.** The start date is the date that will be artificially generated and included in the construction of the event record. *The default behavior is to generate a start date of 5 September 2006.*
- g. **Event Delay.** This item specifies an arbitrary delay between successive generated events. Users wishing to use this feature will need to experiment with values to achieve the desired delay. *The default behavior is to generate events with a 2000 arbitrary delay.*
- f. **Protocol.** The Snare Generator is able to send events either via TCP or UDP, since the Snare Server can collect via any of these means. Windows, IRIX, Universal Log, IIS Web Logs and Solaris events may therefore be sent via UDP or TCP using this selection. SYSLOG events are only able to be sent via the SYSLOG Port 514 (UDP only), so this setting is disregarded for these 2 types of events. *The default behavior is to generate events via UDP.*

3.2 OPERATION

Once the configuration of the SNARE Generator has been completed, the operation (ie. creating and sending events) may begin. Figure 3 shows the front panel of the SNARE Generator during operation.

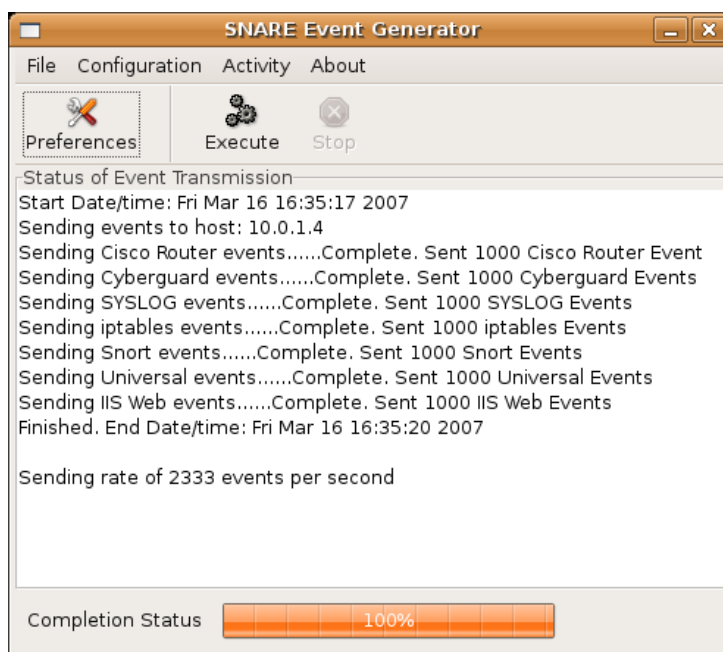


Figure 3 SNARE Generator Operation

In order to commence operation, the 'Execute' button is selected either via the toolbar, or via the 'Activity' menu item. Once this is done, the main view window will show the status of the tasks, and the progress bar at the bottom of the screen (as shown in Figure 3) will show the progress as a percentage of *all* the tasks to be completed. An operation may be terminated at any stage by the user by selecting the 'Stop' button. If this is undertaken, a 'Transmission stopped by the user' message will appear on the main view screen, and the progress bar will freeze at the point at which the stop command was issued.

Also, note that the events per second is also shown in the main window. This is *total* of all events sent by the user.

4 SNARE SERVER



The SNARE Server collects events and logs from a variety of operating systems, applications and appliances including, but not limited to: Windows NT/2000/XP/2003, Solaris, AIX, Irix, Linux, Tru64, ACF2, RACF, CISCO Routers, CISCO PIX Firewall, CyberGuard Firewall, Checkpoint Firewall1, Gauntlet Firewall, Netgear Firewall, IPTables Firewall, Microsoft ISA Server, Microsoft IIS Server, Lotus Notes, Microsoft Proxy Server, Apache, Squid, Snort Network Intrusion Detection Sensors, IBM SOCKS Server, and Generic Syslog Data of any variety.

In addition to the above, the benefits of purchasing the SNARE Server include:

- Official support mechanism for the SNARE open source agents. Note that official SNARE agent support is not offered through *any* other channels.
- All future SNARE Server versions and upgrades included as part of an annual maintenance fee.
- Ability to collect any arbitrary log data, either via UDP or TCP protocols.
- Proven technology that works seamlessly with the SNARE agents.
- SNARE reflector technology that allows for all collected events to be sent, in real time, to a standby/backup SNARE Server.
- Ability to continuously collect large numbers of events. SNARE Server collection rates exceed 60,000 events per minute using a low end, workstation class, Intel based PC on a 100Mbps network.
- Automatic archiving of events to compressed text format after a configurable event time period. This is to prevent the database from slowing down due to storage of old events.
- Ability to drill down from top level reports. This reduces the amount of data “clutter” and allows a system administrator to fine tune the reporting objectives.
- Ability to create “cloned” objectives that allow very specific reporting against any collection profile. These reports, along with all SNARE Server objectives, may be scheduled and emailed to designated staff.
- The SNARE Server uses extensive discriminators for each objective, allowing system administrators to finely tune reporting based on inclusion or exclusion of certain parameters.
- Very simple, single CD installation for those users not requiring a hardware based appliance.

The SNARE Server uses a hardened version of the Linux operating system base for stability and its ability to use a myriad of stable and functional open source tools. A SNARE Server user, however need not be concerned with managing a Linux server. The SNARE Server, once installed, is a fully contained appliance, and does not require any system administrator level maintenance. The SNARE Server will operate on commonly available Intel based PCs, with hardware specifications shown on the next page.

There are supported versions of the SNARE agents which are only available through the purchase of a SNARE Server. Functionality includes, but is not limited to, ability to send events via TCP as well as UDP, and the ability to send events to many destinations, not just one host.

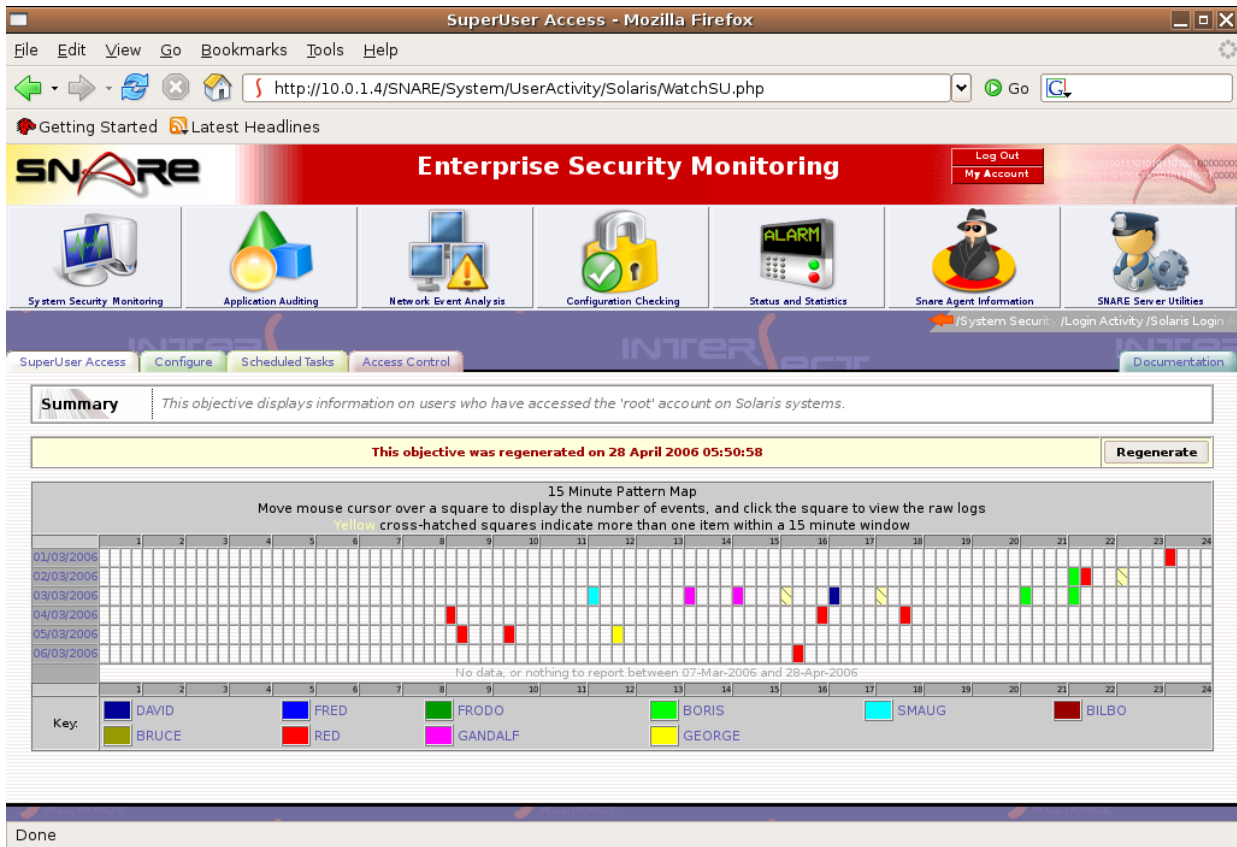


Figure 4: Screen shot from the SNARE Server

5 ABOUT INTERSECT ALLIANCE



Intersect Alliance is a team of leading information technology security specialists. In particular, Intersect Alliance are noted leaders in key aspects of IT Security, including host intrusion detection. Our solutions have and continue to be used in the most sensitive areas of Government and business sectors.

The Intersect Alliance business strategy includes demonstrating our commitment and expertise in IT security by releasing open source products such as SNARE, and the proprietary SNARE Server. Intersect Alliance intend to continue releasing tools that enable users, administrators and clients worldwide to achieve a greater level of productivity and effectiveness in the area of IT Security, by simplifying, abstracting and/or solving complex security problems.

Visit the Intersect Alliance website for more information at www.intersectalliance.com.

6 APPENDIX A – EVENT LOG FORMATS

The SNARE Generator contains a number of 'dummy events' which are used to create the event logs. These 'dummy events' contain tokens, which are replaced at execution time, in order to create the full event logs. These tokens represent variables such as time, date, users, etc.

The following sections describe the events which are generated by the SNARE Generator. The tokens within these 'dummy events' are shown below:

- AAAAAAAAAAAA = Source IP address
- BBBBBBBBBBBB = Destination IP address
- EE FFF GGGG = Day Month Year
- QQQ = Process specifier
- RRRRRRRRR = User Account specifier
- TTTT = Destination port
- WWWWWWW = File path specifier
- XX:YY:ZZ = Time
- ZZZZZZ = User

PIX Firewall Logs

```
char pix_firewall_event_1[512] = "<163>FFF EE GGGG XX:YY:ZZ: %PIX-2-106006: Deny inbound
UDP from AAAAAAAAAAAAA/12345 to BBBBBBBBBBBBB/TTTT on interface outside";
```

```
char pix_firewall_event_2[512] = "<163>FFF EE GGGG XX:YY:ZZ: %PIX-2-106001: Inbound TCP
connection denied from IP_addr/port to IP_addr/port flags TCP_flags on interface int_name Inbound
TCP connection denied from AAAAAAAAAAAAA/9876 to BBBBBBBBBBBBB/TTTT flags SYN on interface
outside";
```

```
char gauntlet_firewall_event[512] = "<163>GGGG-FFF-EE XX:YY:ZZ George_Test kern.info gfw
AAAAAAAAAAAA 2152 BBBBBBBBBBBBB TTTT udp drop securityalert: udp if=qfe0 from
AAAAAAAAAAAA:1234 to BBBBBBBBBBBBB on unserved port TTTT";
```

Solaris BSM Logs

```
char solaris_process_event[512] = "solaris_cora SolarisBSM 1
header,146,2,execve(2),,Day FFF EE XX:YY:ZZ GGGG, + 140001416 msec path,/usr/bin/QQQ
attribute,100555,ZZZZZZ,bin,136,379861,0 exec_args,2,grep,snare
subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3 return,success,0
sequence,65941";
```

```
char solaris_ftp_login_event[512] = "solaris_cora SolarisBSM 1 header,146,2,ftp
access,,Day FFF EE XX:YY:ZZ GGGG, + 140001416 msec
```

```

subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3      return,success,0
sequence,65942";

char solaris_login_telnet_event[512] = "solaris_cora SolarisBSM 1 header,146,2,login
- telnet,,Day FFF EE XX:YY:ZZ GGGG, + 140001416 msec
subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3      text,successful
login return,success,0 sequence,65943";

char solaris_logout_event[512] = "solaris_cora SolarisBSM 1 header,146,2,logout,,Day FFF EE XX:YY:ZZ GGGG, + 140001416 msec
subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3      text,sshd logout
ZZZZZZreturn,success,0 sequence,65944";

char solaris_su_event[512] = "solaris_cora SolarisBSM 1 header,146,2,su,,Day FFF
EE XX:YY:ZZ GGGG, + 140001416 msec
subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3      return,success,0
sequence,6594555";

char solaris_file_event[512] = "solaris_cora SolarisBSM 1 header,146,2,open(2) -
read,,Day FFF EE XX:YY:ZZ GGGG, + 140001416 msec path,/usr/bin/WWWWWWWWW
attribute,100555,ZZZZZZ,bin,136,379861,0
subject,ZZZZZZ,ZZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3      return,success,0
sequence,65946";

```

Windows Logs

```

char windows_login_event[512] = "Windows_Host MSWinEventLog 0 Security
3027 Day FFF EE XX:YY:ZZ GGGG 528 Security ZZZZZZ User Success
Audit Test_Host Successful Logon: User Name: ZZZZZZ Domain: ASH Logon ID:
(0x0,0x1234) Logon Type: 2 Logon Process: User32 Authentication Package: Negotiate Workstation
Name: ASH";

char windows_logoff_event[512] = "Windows_Host MSWinEventLog 0 Security
3028 Day FFF EE XX:YY:ZZ GGGG 538 Security ZZZZZZ User Success
Audit Test_Host User Logoff: User Name: ZZZZZZ Domain: COAL Logon ID:
(0x0,0x1BAE6) Logon Type: 3";

char windows_process_event[512] = "Windows_Host MSWinEventLog 0 Security
3029 Day FFF EE XX:YY:ZZ GGGG 592 Security ZZZZZZ User Success
Audit Test_Host A new process has been created: New Process ID: 2166844768
Image File Name: \\WINNT\system32\wbem\QQQ.exe Creator Process ID: 2166956512 User Name:
ZZZZZZ Domain: FIREBIRD Logon ID: (0x0,0x3E7)";

char windows_file_event[512] = "Windows_Host MSWinEventLog 0 Security
3030 Day FFF EE XX:YY:ZZ GGGG 560 Security ZZZZZZ User Success Audit
Test_Host Object Open: Object Server: Security Object Type: File Object Name:
C:\Directory\WWWWWWWWW.doc New Handle ID: 1220 Operation ID: {0,117792} Process ID: 924
Primary User Name: ZZZZZZ Primary Domain: FLAME Primary Logon ID: (0x0,0x8F9F) Client User
Name: - Client Domain: - Client Logon ID: - Accesses SYNCHRONIZE ReadData (or ListDirectory)
Privileges -";

```

```

char windows_acct_create_event[512] = "Windows_Host  MSWinEventLog  0
Security      3031  Day FFF EE XX:YY:ZZ GGGG  624  Security      ZZZZZZ  User
Success Audit Test_Host                      User Account Created: New Account Name: RRRRRRRRR
New Domain: ASH New Account ID: testuser2 Caller User Name: ZZZZZZ Caller Domain: ASH Caller
Logon ID: (0x0,0x7798) Privileges -";

```

```

char windows_acct_delete_event[512] = "Windows_Host  MSWinEventLog  0
Security      3032  Day FFF EE XX:YY:ZZ GGGG  630  Security      ZZZZZZ  User
Success Audit Test_Host                      User Account Deleted: Target Account Name: RRRRRRRRR
Target Domain: ASH Target Account ID: Unknown (S-1-5-21-1343024091-507921405-1801674531-1001)
Caller User Name: ZZZZZZ Caller Domain: ASH Caller Logon ID: (0x0,0x7798) Privileges: -";

```

```

char windows_group_delete_event[512] = "Windows_Host  MSWinEventLog  0
Security      3033  Day FFF EE XX:YY:ZZ GGGG  638  Security      ZZZZZZ  User
Success Audit Test_Host                      Security Enabled Local Group Deleted: Target Account
Name: RRRRRRRRR Target Domain: LE-I4RFROD15WSP Target Account ID: %S-1-5-21-1844237615-
842925246-1343024091-1002} Caller User Name: ZZZZZZ Caller Domain: LE-I4RFROD15WSP
Caller Logon ID: (0x0,0x9639) Privileges: -";

```

```

char windows_group_create_event[512] = "Windows_Host  MSWinEventLog  0
Security      3034  Day FFF EE XX:YY:ZZ GGGG  635  Security      ZZZZZZ  User
Success Audit Test_Host                      Security Enabled Local Group Created: New Account
Name: RRRRRRRRR New Domain: LE-I4RFROD15WSP New Account ID: %S-1-5-21-1844237615-
842925246-1343024091-1002} Caller User Name: ZZZZZZ Caller Domain: LE-I4RFROD15WSP
Caller Logon ID: (0x0,0x9639) Privileges: -";

```