

SNARE

System iNtrusion Analysis & Reporting Environment

SNARE Generator User Manual

INTERSECT
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

© 1999-2005 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 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.....	6
3.1 Configuration.....	6
3.2 Operation.....	7
4 About InterSect Alliance.....	9
5 Appendix A – Event Log Formats.....	10

1 Introduction



The SNARE Generator has been designed to artificially generate events and send them over the network to a SNARE Sever. As at version 1.2 of the SNARE Generator release, five log types are able to be artificially generated, namely: Windows events, Solaris events, IRIX events, SYSLOG and PIX (CISCO firewall) events. This tool was originally designed to test the SNARE Server operation, but may be useful to parties who use the SNARE Micro Server and generic SYSLOG collection servers.

The artificially generated Windows, IRIX and Solaris events will be sent over the network to UDP/TCP port 6161 or SYSLOG (UCP) port 514, the listening port for the SNARE Server. SYSLOG and PIX events will be sent to the SYSLOG port, namely UPD 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 binary, simply download the executable RPM, and run the command (as root):

rpm -Uvh snare-generator-1.2.rpm

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.

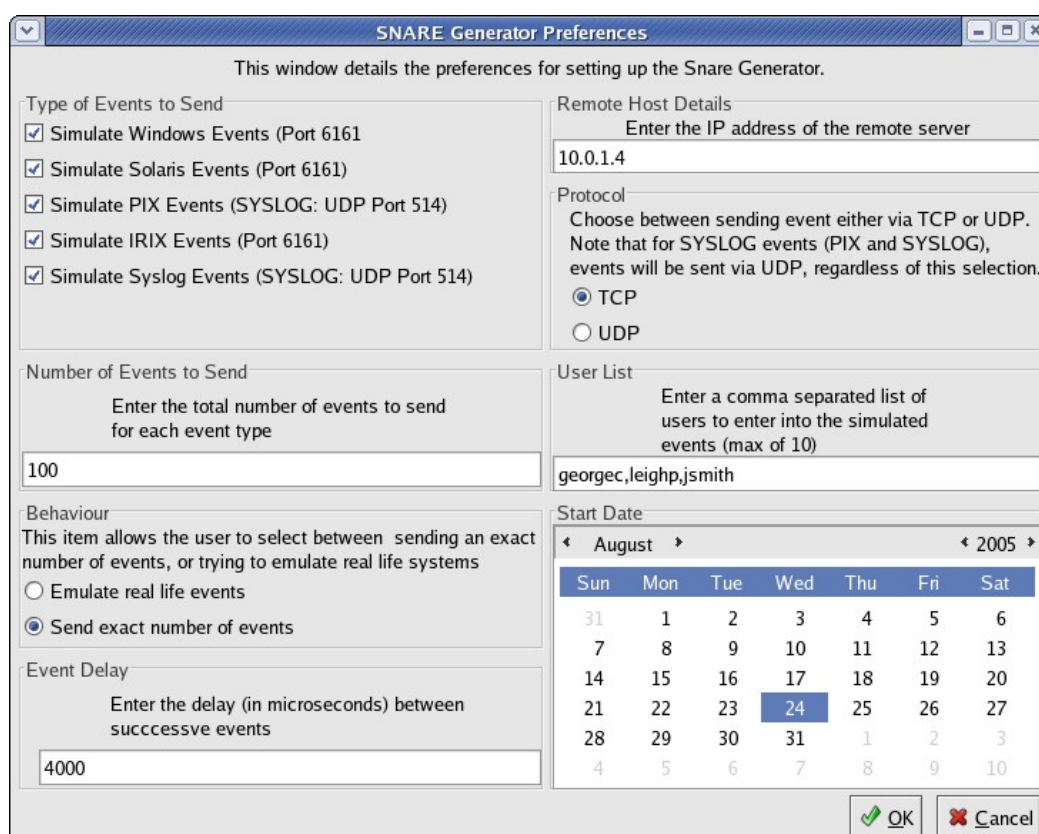


Figure 1 SNARE Generator Main Window

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.



This window details the preferences for setting up the Snare Generator.

Type of Events to Send

- Simulate Windows Events (Port 6161)
- Simulate Solaris Events (Port 6161)
- Simulate PIX Events (SYSLOG: UDP Port 514)
- Simulate IRIX Events (Port 6161)
- Simulate Syslog Events (SYSLOG: UDP Port 514)

Remote Host Details

Enter the IP address of the remote server
10.0.1.4

Protocol

Choose between sending event either via TCP or UDP. Note that for SYSLOG events (PIX and SYSLOG), events will be sent via UDP, regardless of this selection.

TCP
 UDP

Number of Events to Send

Enter the total number of events to send for each event type
100

User List

Enter a comma separated list of users to enter into the simulated events (max of 10)
georgec,leighp,jsmith

Behaviour

This item allows the user to select between sending an exact number of events, or trying to emulate real life systems

Emulate real life events
 Send exact number of events

Event Delay

Enter the delay (in microseconds) between successive events
4000

Start Date

← August → ← 2005 →

Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3
4	5	6	7	8	9	10

OK Cancel

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.

- Types of Events to Send.** As at Version 1.2, there are five log types which may be generated, namely Windows logs, IRIX logs, Solaris (BSM) logs, SYSLOG and PIX logs. 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 and Solaris event types, events are sent to the default SNARE Server port of 6161, either via UDP or TCP. For the SYSLOG and PIX firewall logs, 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 deselecting the relevant checkbox as shown in Figure 2. *The default behavior is to turn on all five 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 100 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.** The SNARE Generator was originally developed to simulate the 'real world' event logging environments seen in various locations. This feature allowed events to be generated and sent based on the probability of an event being generated, that depended directly on the time of day. For example, if the 'Emulate real life events' radio button is checked, then the SNARE Generator will generate PIX firewall events (for example) at a probability of 75% between 5pm and 10am, and 3pm and 10pm. At other times, the event probability will be 95%. This feature also applies to Windows and Solaris event generation, though the probabilities will vary greatly to the example given. If this type of behavior is required, then note that the 'Number of Events to Send' item will be an approximate figure. If the 'Send exact number of events' radio button is checked, then the type of behavior described above is ignored and one type of each event record sub-type will be generated every 'second'. The 'second' time designator does not refer to the real-time, but rather to the time that is artificially generated when the event record is created.

Each event log type generated by the SNARE Generator has a number of sub-types. These sub-types along with the relevant 'tokens' that are used to construct the event record are shown in Appendix A - Event Log Format. *In almost all cases, it is expected that the default behavior of 'Send exact number of events' will be used.*

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 24 August 2005.*

g. **Event Delay.** This item specifies the delay (in microseconds) between successive generated events. The delay is specified in microseconds. *The default behavior is to generate events with 4msec (4000 micro secs) 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 and Solaris events may therefore be sent via UDP or TCP using this selection. SYSLOG and PIX 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 TCP.*

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.

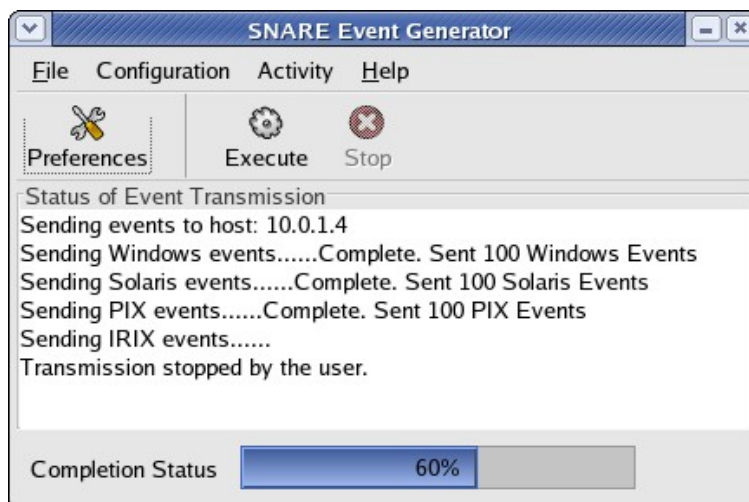


Figure 3 SNARE Generator Operation

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.

4 SNARE Server

The team at Intersect Alliance have produced software that enables remote control, collection, analysis and of output from all SNARE agents, including Windows, Solaris, Linux and IRIX, as well as applications such as web servers and appliances that generate SYSLOG formatted events. This software is known as the SNARE Server, and full details are available from the Intersect Alliance web site (www.intersectalliance.com). The SNARE Server is proprietary software, and is not available as open source.

The SNARE Server is an Enterprise Audit Event Log analysis solution, comprising a central audit event collection, analysis, reporting and archive service, and security 'agents' for multiple operating systems and applications.

Full source code and documentation is provided with this service, allowing the InterSect Alliance partners, or internal security professionals, to quickly develop SNARE security objectives that are derived directly from your key organisational risk items. The SNARE Server also comes equipped by default with an array of security objectives that allows agencies to meet common security goals. A selected screen shot of the SNARE Server is shown below in Figure 4. Full details on the SNARE Server, including more screen shots are available from the Intersect Alliance web site.

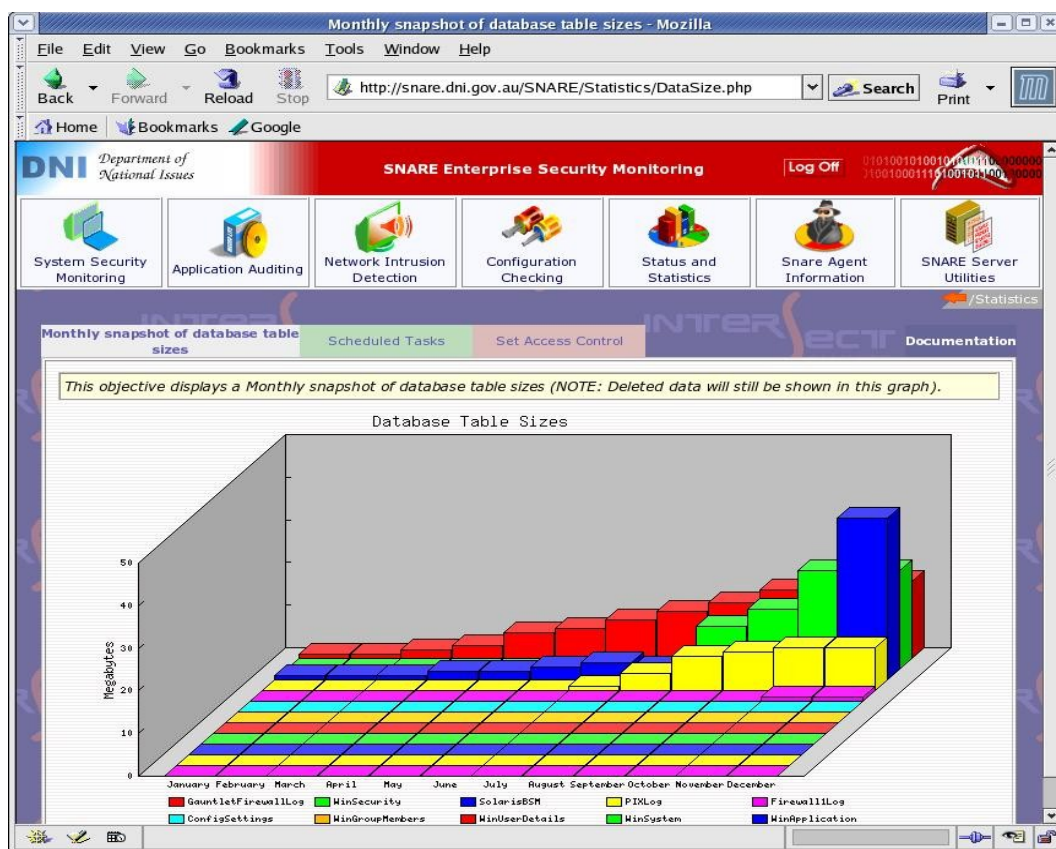


Figure 4: Screen shot from the SNARE Server

5 About InterSect Alliance



InterSect Alliance is a team of leading information technology security specialists in both the 'technical' and 'policy' areas. 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. Intersect Alliance consult and contract to a number of agencies in Australia and the Asia Pacific, for both the business and Government sectors.

The Intersect Alliance business strategy includes demonstrating our commitment and expertise in IT security by releasing Open Source products such as SNARE. 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
 RRRRRRRRRR = User Account specifier
 TTTT = Destination port
 WWWWWWWW = 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 BBBBBBBBBBBB/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 BBBBBBBBBBBB/TTTT flags SYN on interface outside";
```

```
char gauntlet_firewall_event[512] = "<163>GGGG-FFF-EE XX:YY:ZZ George_Test kern.info gfw AAAAAAAAAAAAA 2152 BBBBBBBBBBBB TTTT udp drop securityalert: udp if=qfe0 from AAAAAAAAAAAAA:1234 to BBBBBBBBBBBB 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,ZZZZZ,bin,136,379861,0 exec_args,2,grep,snare subject,ZZZZZ,ZZZZZ,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,ZZZZZ,ZZZZZ,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,ZZZZZ,ZZZZZ,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,ZZZZZ,ZZZZZ,other,root,other,12228,12212,8236 131095 10.0.1.3 text,sshd logout
ZZZZZ return,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,ZZZZZ,ZZZZZ,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,ZZZZZ,bin,136,379861,0
subject,ZZZZZ,ZZZZZ,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 ZZZZZ User
Success Audit Test_Host Successful Logon: User Name: ZZZZZ 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 ZZZZZ User
Success Audit Test_Host User Logoff: User Name: ZZZZZ 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 ZZZZZ
User Success Audit Test_Host A new process has been created: New Process
ID: 2166844768 Image File Name: \\WINNT\system32\wbem\lqqq.exe Creator Process ID:
2166956512 User Name: ZZZZZ 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 ZZZZZ 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: ZZZZZ 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-I4RFR0D15WSP Target Account ID: %{S-1-5-
21-1844237615-842925246-1343024091-1002} Caller User Name: ZZZZZZ Caller Domain: LE-
I4RFR0D15WSP 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-I4RFR0D15WSP New Account ID: %{S-1-5-21-
1844237615-842925246-1343024091-1002} Caller User Name: ZZZZZZ Caller Domain: LE-
I4RFR0D15WSP Caller Logon ID: (0x0,0x9639) Privileges: -";

```