



Nexpose Integration Guide for Lieberman Enterprise RPM

Enabling Credential Access for Nexpose Scanning

Partner Name: Lieberman

Website: <http://www.liebsoft.com>

Product Name: Lieberman Enterprise Random Password Manager

Version: 4.83.7

Action Type: Automated via API and Web SDK URL



Solution Summary

Random Password Manager is designed to randomize and store the passwords for accounts on target systems on a regular recurring basis. Because these passwords are stored and managed by the program, they can be retrieved via a delegated web interface. Access to the password store as well as other web interface features can be limited to specific Windows groups, Windows users, or explicit accounts

Partner Product Configuration

The Web Application component should be installed with the SDK URL option activated and running through a SSL tunnel. Please see the “Manage Web Application” section under the Lieberman online ERPM guide for installation and configuration instructions, or Page 194 of the ERPM Administrator Guide.

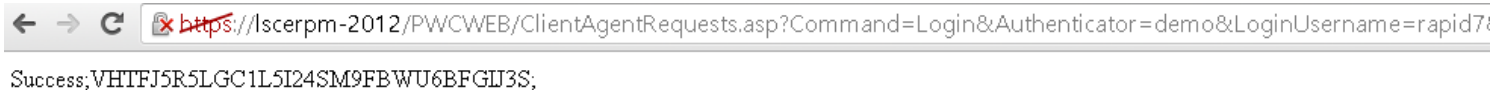
Once installed and configured, the Web App SDK URL should be accessible through:

<https://lieberman-erpm-ip/PWCWEB/ClientAgentRequest.asp>

As such, opening a browser and pointing it to the following URL should return “Success”

[https://lieberman-erpm-ip/PWCWEB/ClientAgentRequest.asp?Command=Login&Authenticator=\\$DOMAIN&LoginUsername=\\$USERNAME&LoginPassword=\\$PASSWORD](https://lieberman-erpm-ip/PWCWEB/ClientAgentRequest.asp?Command=Login&Authenticator=$DOMAIN&LoginUsername=$USERNAME&LoginPassword=$PASSWORD)

(Please replace \$DOMAIN, \$USERNAME, \$PASSWORD with the appropriate information)

A screenshot of a web browser window. The address bar shows a URL starting with 'https://lscerpm-2012/PWCWEB/ClientAgentRequests.asp?Command=Login&Authenticator=demo&LoginUsername=rapid7'. Below the address bar, the page content displays 'Success;VHTFJ5R5LGC1L5I24SM9FBWU6BFGIJ3S;'.

Success;VHTFJ5R5LGC1L5I24SM9FBWU6BFGIJ3S;

Also, keep in mind that by default, ERPM only allows a max amount of password check-outs per user, this setting is configurable through the ERPM “Manage Web App” interface:

Web Application Settings - LSCERPM-2012

Remote Sessions		Console Display		User Dashboards	
App Options	Password Access	File Repository Settings	Account Elevation	Security	User/Session Management
<input checked="" type="checkbox"/> Enable password check-out					
Default check-out/extension durations (minutes)					
Windows Accounts:	120	Linux/Unix Accounts:	120		
AS400 Accounts:	120	OS390 Accounts:	120		
IPMI Accounts:	120	DRAC Accounts:	120		
Cisco Accounts:	120	SQL Server Accounts:	120		
Oracle Accounts:	120	Sybase Accounts:	120		
MySQL Accounts:	120	LDAP Accounts:	120		
Shared Account Lists:	120				
<input type="checkbox"/> Require recovery comment for password recoveries					
<input type="checkbox"/> Require ticket number for password recoveries					
<input type="checkbox"/> Verify ticket number with BMC Remedy IT Service Manager					
<input type="checkbox"/> Verify ticket number with Microsoft System Center Service Manager					
<input type="checkbox"/> Verify ticket number with HP Service Manager					
<input type="checkbox"/> Verify ticket number with ServiceNow					
Password request timeout window (minutes):				60	
Password request grant timeout window (minutes):				60	
<input type="checkbox"/> Allow users to edit and delete managed random passwords					
Maximum password check-out duration (minutes):				720	
Maximum simultaneous check-outs:				8	
<input type="checkbox"/> Block password check-in if password is in use					
<input type="checkbox"/> Log account in use on check-in to system's event log				LSCERPM-2012	
<input type="checkbox"/> Log all password check-outs to system's event log				LSCERPM-2012	
<input type="checkbox"/> Log all password check-ins to system's event log				LSCERPM-2012	
<input type="checkbox"/> Allow users to request check-outs in the future					
Password request window for future check-outs (hours):				0	
<input type="checkbox"/> Block password check-out if password spin job creation fails					
<input type="checkbox"/> Allow users to check out passwords to any group they are a member of					
<input type="checkbox"/> Require check-in comment when password is checked-in					

Please refer to your Lieberman Administrator Guide for fine tuning these options.

Finally, under the "Security" tab in the Web Application Settings window, make sure "Enable Windows Authentication (must be enabled in IIS)" is unchecked:

Web Application Settings - LSCERPM-2012

Remote Sessions		Console Display		User Dashboards	
App Options	Password Access	File Repository Settings	Account Elevation	Security	User/Session Management
<input type="checkbox"/> Allow default authenticated user access					
<input checked="" type="checkbox"/> Hide recovered password after (seconds):				90	
Force inactive web session timeout (minutes):				20	
<input type="checkbox"/> Require secure cookies (requires SSL enabled for the site)					
<input type="checkbox"/> Enable Windows Integrated Authentication (must be enabled in IIS)					
<input type="checkbox"/> Automatically login users using Windows Integrated Authentication					

If configuration is an issue, please contact your Lieberman representative for support.

Introduction

This document will guide you through all the steps necessary to configure the Lieberman script to successfully import Lieberman credentials into the Nexpose vulnerability management system.

Before you begin

The script was created using Ruby, as such, a Ruby interpreter must be installed on the system where it's going to run. The following link shows the different options for installing Ruby in several platforms:

- <https://www.ruby-lang.org/en/downloads/>

Please install the most appropriate for your need.

Once installed, the following Ruby Gems must also be installed:

- nexpose (<http://rubygems.org/gems/nexpose>)

In some systems, dependencies might need to be installed prior to the installation of each Gem. Please refer to their appropriate documentation for instructions.

Configuring the script

Once all dependencies have been installed, the script can now be configured. To configure, open the script with a text editor (Notepad, Notepad++, etc)

- Configure Nexpose settings:

```
# NEXPOSE CONFIGURATION

# Nexpose username
# Example: @@nxuser = 'nxadmin'
@@nxuser = 'nxadmin'

# Nexpose password
# Example: @@nxpass = 'nxadmin'
@@nxpass = 'nxadmin'

# Nexpose console IP address
# Example: @@console = '127.0.0.1'
@@console = '127.0.0.1'
```

- A valid username, password and console IP address. We recommend creating a user with permissions to create reports on the sites/asset groups necessary.

- Configure Logging settings:

```
# LOGGING.
# This script includes a logger, all output will be sent to the file service_now.log in the director
# where this script is run.
require 'Logger'
$LOG = Logger.new('lieberman.log', 'monthly')

# Valid log levels: Logger::DEBUG Logger::INFO Logger::WARN Logger::ERROR Logger::FATAL
$LOG.level = Logger::INFO
```

- A logger is included with the script that outputs by default all INFO events to the file service_now.log in the directory where the script is run. To configure this setting, feel free to change the logging level or the name of the file to be output in this setting.

➤ Configure Lieberman settings:

```
# Lieberman Web SDK URL.
@@lieberman_instance = "https://lscerpm-2012/PWCWEB/ClientAgentRequests.asp"
# Lieberman Authenticator domain.
@@lieberman_authenticator = "demo"
# Lieberman username.
@@lieberman_user = "rapid7"
# Lieberman password
@@lieberman_password = "abc"
```

➤ Run the script for the first time.

- The script can be run using the command from the command line:


```
> ruby lieberman_integration.rb
```
- While running, all logging information will be output to the lieberman.log file.
- Once finished, the sites should have updated credentials for the systems:

Site Configuration

Previous

Next

Save

Cancel

General

Assets

Scan Setup

Alerting

Credentials

Web Applications

Organization

Access

Create or edit credentials for authenticated scans that will be used specifically for this site. You can only edit site-specific credentials in a site configuration.

Credential Listing

Name	Scope	Use in Scans	Service	Domain	User name
Imported from API	Site-specific	✓	Secure Shell (SSH)	[Linux]	root
Imported from API	Site-specific	✓	Microsoft Windows/Samba (SMB/CIFS)	LSCERPM-2012	Administrator
Imported from API	Site-specific	✓	Microsoft Windows/Samba (SMB/CIFS)	LSCTOOLS-2012	Administrator
Imported from API	Site-specific	✓	Microsoft Windows/Samba (SMB/CIFS)	DB2008-SQL08R2	Administrator
Imported from API	Site-specific	✓	Secure Shell (SSH)	[Linux]	root
Imported from API	Site-specific	✓	Microsoft Windows/Samba (SMB/CIFS)	WEB2K8R2-WS	Administrator
Imported from API	Site-specific	✓	Microsoft Windows/Samba (SMB/CIFS)	DEMO	Administrator
Imported from API	Site-specific	✓	Secure Shell (SSH)	[Linux]	root

Recommendations

We recommend running the script after creating the sites. Due to the Lieberman SDK nature only hostnames can be queried for credentials (No IP addresses), if a FQDN (asset.domain.com) is specified in the site listing the script will only use the name before the first period (asset).

This script will also use the first account Lieberman returns; in effect, if there are two accounts being managed for one asset in Lieberman, only the first one will be used.

We recommend creating local administrator accounts on the systems and have them managed by Lieberman.

What if something goes wrong?

The most common errors when running the script are configuration based, users without permission to update sites, or query credentials from Lieberman.

1. - Make sure we can query Lieberman ERPM for a credential; use the Login test to validate that the API is available, open a browser tab to:

[https://lieberman-erpm-ip/PWCWEB/ClientAgentRequest.asp?Command=Login&Authenticator=\\$DOMAIN&LoginUsername=\\$USERNAME&LoginPassword=\\$PASSWORD](https://lieberman-erpm-ip/PWCWEB/ClientAgentRequest.asp?Command=Login&Authenticator=$DOMAIN&LoginUsername=$USERNAME&LoginPassword=$PASSWORD)

(Replacing the \$DOMAIN, \$USERNAME, \$PASSWORD) variables. It should return "Success; XXXXX" where XX is a string of characters.

2. - Browse through the Lieberman.log, in it we explain a set of different situations:

a. "Could not retrieve account information" means that Lieberman did not provide account information for that asset, it's either probably not managed in Lieberman or we don't have permissions to retrieve it.

b. "Could not retrieve password for XXX in Lieberman. Check your permissions." Means that either the user used cannot retrieve those credentials, or the max amount of check-outs have been reached. Please contact your Lieberman support representative.

3. - If anything else fails, please email us to integrations_support@rapid7.com with the log and information about the issue.