RSA Archer
Weekly Free Friday Webcast:

Maintenance, Monitoring, and Alerting

Jeff Letterman – Technical Support Engineer III

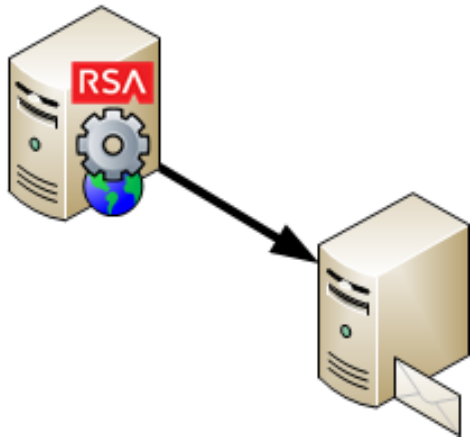
March 21, 2014

Contact Information
Email: archersupport@rsa.com

Test Email Notifications

- Need a quick way to confirm an Archer Server can send email notifications to the SMTP Server specified in the Archer Control Panel (ACP)?
- Use PowerShell to send a test email to the SMTP Server

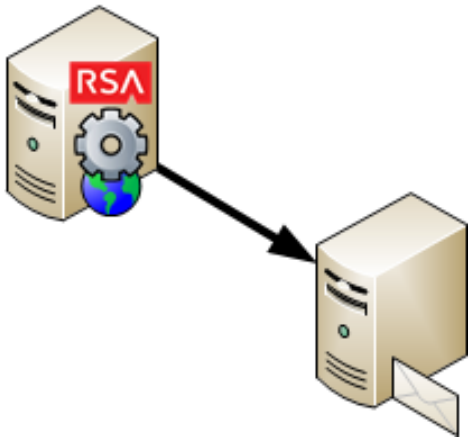
```
#Send Test email
$MailMessage = @{
    To = 'ArcherAdmin@YourCompany.com'
    From = 'ArcherServiceMonitor@YourCompany.com'
    Subject = 'Test Email using PowerShell'
    Body = 'Hello'
    Smtpserver = '127.0.0.1'
    ErrorAction = 'SilentlyContinue'
}

Send-MailMessage @MailMessage
```



- smtp4dev is a tool used for testing email notifications
  - http://smtp4dev.codeplex.com/

Demo Test Email

- Tools used in demonstration
  - Windows PowerShell Modules › Run as Administrator
  - smtp4dev
- Perform on Archer Servers



Maintenance

- Need a nice and easy way to notify users the Archer web site is being upgraded to a new Archer version, or installing an Archer Package, or performing other site maintenance?
- Create or move a file named app_offline.htm into the Web site's virtual directory
  - \inetpub\wwwroot\Archer
- Little known IIS feature starting with ASP.NET 2.0
- All requests to the Web site are redirected to app_offline.htm
- Add custom message with information about the outage, change control, completion time, point-of-contact, etc.

- When maintenance is complete, remove the app_offline.htm file to bring the Web site online again

Maintenance (continued)

- Images must reference another site's image or embed in Base64 format
- Base64 image converters are available online
  - http://www.askapache.com/online-tools/base64-image-converter/
- Base64 Example and Image

R0lGODlhDwAPAKECAAAAzMzM/////wAAACwAAAAADwAPAAACIISPeQHsrZ5ModrLl
N48CXF8m2iQ3YmmKqVlRtW4MLwWACH+H09wdGltaXplZCBieSBVbGVhZCBTbWFy
dFNhdmVyIQAAOw==

Maintenance (continued)

- Simple example of the app_offline.htm file

```
<html>
  <head>
  </head>
  <body style="font-size: xx-large; font-family: Arial; text-align: center">
    <div>
      <p>The Archer web site is under maintenance.</p>
      <img alt="Base64 Image"
src="
APAAACIISPeQHsrZ5ModrLlN48CXF8m2iQ3YmmKqVlRtW4MLwWACH+H09wdGltaX
plZCBieSBVbGVhZCBTbWFydFNhdmVyIQAAOw==" />
    </div>
  </body>
</html>
```

Demo app_offline.htm

- Tools used in demonstration
  - Windows Explorer
  - Notepad
  - Internet Explorer
- Perform on Archer Web Servers

WMI

- Need a way to be notified when an Archer service stops/starts or its configuration changes?
- Use Windows Management Instrumentation (WMI) and create a Permanent Event Consumer that subscribes to an event and alerts via email notification
- WMI runs as a Windows service and provides access to a series of “databases”
- WMI Namespace is like a database
  - root, root\subscription, root\cimv2, root\Microsoft\SqlServer
- WMI Class is like a data table containing Properties (data columns) and Methods (actions)
  - Win32_Service, Win32_Process, __EventFilter
- WMI Instance is like a row of data

WMI (continued)

- Example of a WMI Instance (MOF Text)

```
instance of Win32_Service
{
        AcceptPause = FALSE;
        AcceptStop = TRUE;
        Caption = "RSA Archer Configuration";
        CheckPoint = 0;
        CreationClassName = "Win32_Service";
        Description = "Manages the configuration database, client notification, and retrieval of configuration information.
If this service is stopped, requests for configuration information will not be processed and any instances depending
on this configuration information will fail.";
        DesktopInteract = FALSE;
        DisplayName = "RSA Archer Configuration";
        ErrorControl = "Normal";
        ExitCode = 0;
        Name = "RSAArcherConfigurationService";
        PathName = "\"C:\\Program Files\\RSA Archer\\Services\\ArcherTech.Services.ConfigurationService.exe\"";
        ProcessId = 3212;
        ServiceSpecificExitCode = 0;
        ServiceType = "Own Process";
        Started = TRUE;
        StartMode = "Auto";
        StartName = "LocalSystem";
        State = "Running";
        Status = "OK";
        SystemCreationClassName = "Win32_ComputerSystem";
        SystemName = "ServerName";
        TagId = 0;
        WaitHint = 0;
};
```

WMI (continued)

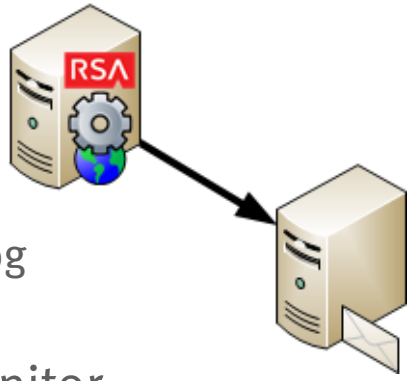
- WMI can monitor for other situations
  - low disk space, processes, high CPU utilization, high memory usage, etc.
- WMI can take other actions
  - run a script, execute a command, restart a service, write to a log file
- WMI can be used for other things
  - data collection, inventory, troubleshooting, etc.
- To test and explore WMI, run wbemtest.exe

Demo WMI

- Tools used in demonstration
  - wbemtest.exe
- Perform on any Windows machine

Monitoring & Alerting

- How to tell WMI what to monitor and send an alert via email?
- Use PowerShell to create the Permanent Event Consumer in WMI
- Create an Event Filter to detect changes to any Archer service
  - SELECT * FROM __InstanceModificationEvent WITHIN 30 WHERE TargetInstance ISA 'Win32_Service' AND Targetinstance.Name LIKE '%Archer%'
- Create an Event Consumer to take action on the Filter
  - SMTPEventConsumer sends emails
  - ActiveScriptEventConsumer executes a script
  - LogFileEventConsumer writes to a log file
  - NTEventLogEventConsumer writes to the Application event log
  - CommandLineEventConsumer launches a process



- Create a Filter to Consumer Binding to “activate” the monitor
  - __InstanceModificationEvent, __InstanceCreationEvent, __InstanceDeletionEvent
- Alternative implementation is to run mofcomp.exe with a Managed Object Format (.MOF) file: mofcomp.exe FileName.mof

Monitoring & Alerting (continued)

- Event Filter – PowerShell Commands

```
#Create a new Event Filter
$instanceFilter = ([wmiclass]"\\.\root\subscription:__EventFilter").CreateInstance()
$instanceFilter.Name = 'Archer Service Monitor Event Filter'
$instanceFilter.EventNamespace = 'root\cimv2'
$instanceFilter.QueryLanguage = 'WQL'
$instanceFilter.Query = "SELECT * FROM __InstanceModificationEvent WITHIN 30 WHERE
TargetInstance ISA 'Win32_Service' AND TargetInstance.Name LIKE '%Archer%' "
$result = $instanceFilter.Put()
$newFilter = $result.Path
```

Monitoring & Alerting (continued)

- Event Consumer – PowerShell Commands

```
#Create a new Event Consumer
$instanceConsumer = ([wmiclass]"\\.\root\subscription:SMTPEventConsumer").CreateInstance()
$instanceConsumer.Name = 'Archer Service Monitor SMTPEventConsumer'
$instanceConsumer.SMTPServer = '127.0.0.1'
$instanceConsumer.ToLine = 'ArcherAdmin@YourCompany.com'
$instanceConsumer.FromLine = 'ArcherServiceMonitor'
$instanceConsumer.Subject = "Archer Service Monitor detected a change to
%TargetInstance.DisplayName%"
$msg = @"
Server:  %TargetInstance.__Server%
Name:    %TargetInstance.DisplayName%
State     was  %PreviousInstance.State%  now  %TargetInstance.State%
Process Id  was  %PreviousInstance.ProcessId%  now  %TargetInstance.ProcessId%
Start Mode  was  %PreviousInstance.StartMode%  now  %TargetInstance.StartMode%
Start Name  was  %PreviousInstance.StartName%  now  %TargetInstance.StartName%
Path Name  was  %PreviousInstance.PathName%  now  %TargetInstance.PathName%
"@
$instanceConsumer.Message = $msg
$result = $instanceConsumer.Put()
$newConsumer = $result.Path
```

Monitoring & Alerting (continued)

- Filter to Consumer Binding – PowerShell Commands

```
#Create a new Binding for the Filter and Consumer
$instanceBinding = ([wmiclass]"\\.\root\subscription:__FilterToConsumerBinding").CreateInstance()
$instanceBinding.Filter = $newFilter
$instanceBinding.Consumer = $newConsumer
$result = $instanceBinding.Put()
$newBinding = $result.Path
```

- To stop monitoring the services, delete the Binding, Consumer, and Filter by appending the following to the PowerShell script

```
([wmi]$newBinding).Delete()
([wmi]$newConsumer).Delete()
([wmi]$newFilter).Delete()
```

Monitoring & Alerting (continued)

- List Event Filters

  Get-WMIObject -Namespace root\Subscription -Class __EventFilter

- List Event Consumers

  Get-WMIObject -Namespace root\Subscription -Class __EventConsumer

- List Event Bindings

  Get-WMIObject -Namespace root\Subscription -Class __FilterToConsumerBinding

- Other PowerShell commands

  Get-WMIObject Win32_LogicalDisk
  Get-WMIObject Win32_LogicalDisk | Select-Object * -excludeproperty "_*"
  Get-WMIObject Win32_Service | Select-Object * -excludeproperty "_*"
  Get-WMIObject Win32_Processor | Select-Object * -excludeproperty "_*"
  Get-WMIObject Win32_ComputerSystem | Select-Object * -excludeproperty "_*"
  Get-WMIObject Win32_OperatingSystem | Select-Object * -excludeproperty "_*"

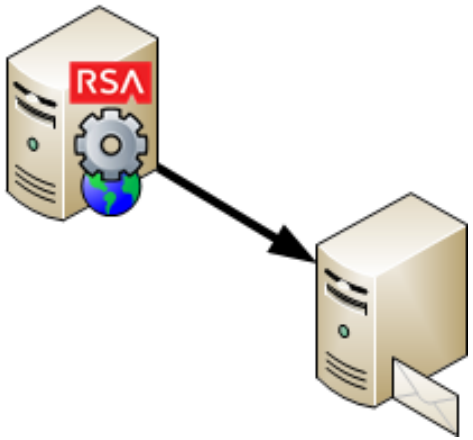
Monitoring & Alerting (continued)

- Sample .MOF File – WMI Format and Alternative to PowerShell

```
// 1. If the WMI repository is rebuilt in the future, the contents of this MOF file will be included in the new WMI repository.
#PRAGMA AUTORECOVER

// 2. Change the context to Root\Subscription namespace.  All standard consumer classes are registered there.
#PRAGMA NAMESPACE("\\\\.\\root\\subscription")

// 3. Create an instance of __EventFilter class and use its Query property to store the WQL event query.
instance of __EventFilter as $instanceFilter
{
    EventNamespace = "Root\\Cimv2";
    Name  = "Archer Service Monitor EventFilter";
    Query = "SELECT * FROM __InstanceModificationEvent WITHIN 30 WHERE TargetInstance ISA \"Win32_Service\" AND Targetinstance.Name LIKE \"%Archer%\" ";
    QueryLanguage = "WQL";
};

// 4. Create an instance of __EventConsumer derived class. (SMTPEventConsumer, ActiveScriptEventConsumer, etc...)
instance of SMTPEventConsumer as $instanceConsumer
{
    Name = "Archer Service Monitor SMTPEventConsumer";
    SMTPServer = "127.0.0.1";
    ToLine = "ArcherAdmin@YourCompany.com";
    FromLine = "ArcherServiceMonitor";
    Subject = "Archer Service Monitor detected a change to %TargetInstance.DisplayName%";
    Message = "Server:       %TargetInstance.__Server%\n"
            "Name:         %TargetInstance.DisplayName%\n"
            "State      was  %PreviousInstance.State%  now  %TargetInstance.State%\n"
            "Process Id  was  %PreviousInstance.ProcessId%  now  %TargetInstance.ProcessId%\n"
            "Start Mode  was  %PreviousInstance.StartMode%  now  %TargetInstance.StartMode%\n"
            "Start Name  was  %PreviousInstance.StartName%  now  %TargetInstance.StartName%\n"
            "Path Name  was  %PreviousInstance.PathName%  now  %TargetInstance.PathName%\n";
};

// 5. Join the two instances by creating an instance of __FilterToConsumerBinding class.
instance of __FilterToConsumerBinding
{
    Consumer   = $instanceConsumer;
    Filter = $instanceFilter;
};
```

Demo Archer Service Monitor

- Tools used in demonstration
  - Windows PowerShell Modules › Run as Administrator
  - smtp4dev
- Perform on Archer Servers



Additional References

- About WMI
  - http://msdn.microsoft.com/en-us/library/aa384642(v=vs.85).aspx
- Monitoring Events
  - http://msdn.microsoft.com/en-us/library/aa392396(v=vs.85).aspx
- An Insider's Guide to Using WMI Events and PowerShell
  - http://blogs.technet.com/b/heyscriptingguy/archive/2012/06/08/an-insider-s-guide-to-using-wmi-events-and-powershell.aspx
- Use PowerShell to Monitor and Respond to Events on Your Server
  - http://blogs.technet.com/b/heyscriptingguy/archive/2010/12/07/use-powershell-to-monitor-and-respond-to-events-on-your-server.aspx
- App_offline.htm
  - http://weblogs.asp.net/scottgu/archive/2005/10/06/426755.aspx
- App_offline.htm gotchas with ASP.NET MVC
  - http://blog.kurtschindler.net/app_offline-htm-gotchas-with-asp-net-mvc/

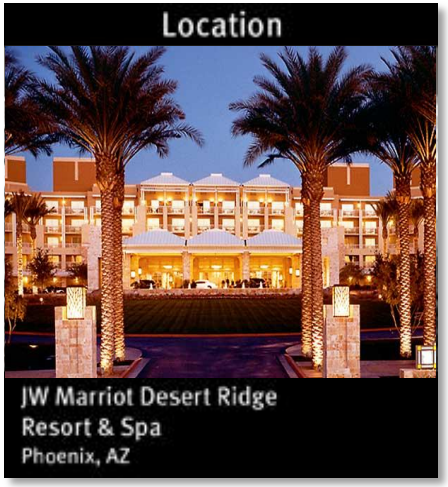
Questions / Issues

- RSA SecurCare Online (SCOL)
  - Archer GRC Summary
    - https://knowledge.rsasecurity.com/scolcms/sets.aspx?product=archer
  - Knowledge Base
    - https://knowledge.rsasecurity.com/scolcms/knowledge.aspx
  - Case Management
    - Login to SCOL › click My Support › click My Products
- Email
  - for Archer product: archersupport@rsa.com
  - for Archer Community or SCOL: support@rsa.com
- Phone
  - USA: 1-800-995-5095 (option 5 for Archer)
  - Global numbers
    - http://www.emc.com/support/rsa/contact/phone-numbers.htm

2014 RSA Archer grc summit

Harnessing Risk.
Exploring Opportunity.
June 10-12
Phoenix, AZ

- 1,000+ GRC professionals
- 42 Customer-led presentations
- 14 Technical breakout sessions
- 12 Working group & roadmap sessions
- Industry roundtables, executive-level breakouts & analyst sessions



Register: rsa.im/grcsummit14

RSA EMC²

- **Summit Award Nominations** will open in a few days
- Check any of the Archer Communities for the **Online Submission Form**:
  - RSA Archer
  - RSA Archer Exchange
  - GRC Ecosystem
- **We WANT to hear from You !**

RSA Archer Course Offerings

System Administrator

elearning
- Introduction to GRC
- Getting Started with the RSA Archer Platform
- Navigating the RSA Archer Platform

Instructor-Led
- RSA Archer Administration
- RSA Archer Advanced Administration

Business Owner

elearning
- Introduction to GRC
- Getting Started with the RSA Archer Platform
- Navigating the RSA Archer Platform
- Archer Solution Fundamentals

Instructor-Led
- RSA Archer Administration
- Getting Started with Enterprise Risk Management
- Getting Started with Policy & Compliance Management

End User

elearning
- Introduction to GRC
- Getting Started with the RSA Archer Platform
- Navigating the RSA Archer Platform

Customized RSA Archer End User training is available upon request

www.emc.com/rsa-training

THANK YOU