QA Checks
User Guide

This document is intended for support staff and engineers that build and maintain servers within your various environments.

This document refers to version 3.17 and above of the scripts

Contents

Overview

The QA checks came about as a need to verify the build of new servers for our various customers and their environments. All new server builds are done with a custom gold image; however this image still lacks many of the additional tools and configuration settings needed before it can be accepted into support.

Most of this extra tooling and configuration is automated, however checks are still needed to make sure each customer has their specific settings in place.

Technical Details

The scripts are written in the Microsoft PowerShell language, with version 2.0 in mind. This is the basic version installed by default on Windows Server 2008 R2.

The script will run on almost all Windows Operating systems, as long as PowerShell version 2.0 or greater is installed, and the PowerShell window is run with administrative privleges.

Supported Operating Systems

Windows Server 2008 R2
Windows Server 2012
Windows Server 2012 R2
Windows Server 2016

Unsupported Operating Systems

Windows 2003 Server
Windows Server 2008
Any non-server operating system

While the scripts will work and produce results on the above list, they are not supported and some scripts may fail.

The Checks

There are almost 90 different checks split over 10 sections. These are executed whenever the QA script is executed against a server and can usually take anywhere between 30 seconds and a couple of minutes to complete. If you are checking multiple servers then this time is per server. The script is set to run up to 5 checks concurrently.

Each check is written to be as efficient as possible; however due to the nature of some of them they may take a little longer than normal. With this in mind, each individual check has a timeout of 60 seconds. This should give them plenty of time to complete their task.

For a full list of checks and sections, they are listed in **Appendix C**.

Quick Start

If you want to see how things turn out, and don't want to start changing setting just yet, simply follow the steps below to produce your first report…

1. Copy the compiled QA script to the target server,
2. On the target server, open an elevated PowerShell window,
3. Change to the folder when the script is saved,
4. If required enter the following command:
   ```
   Set-ExecutionPolicy -ExecutionPolicy Bypass -Force
   ```
5. Execute the QA script file:
   ```
   .\QA_v3.17.xxxx.ps1 -ComputerName localhost
   ```
6. Wait for the script to complete

This will execute the QA scripts and produce a report for you. This is stored in {system-drive}:\QA\Reports.

Customising the settings

To get the most out of the QA scripts, they should be configured for your environment. The default settings are quite strict in terms of security settings and permissions; this is because they were originally designed for an environment with very high security restrictions.

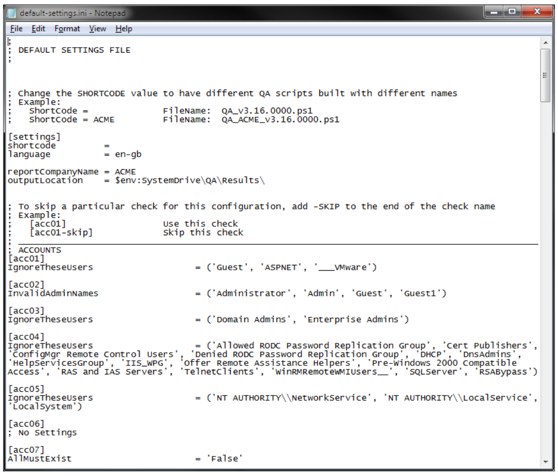
Do not change or save over the "default-setting.ini" file as this will be over written every time you download updates to the QA scripts. Always make a backup before editing.

There are two ways of editing the settings:

1. Manually copy and edit the INI file using notepad or other simple text editor,
2. Use the QA Settings Configurator that was written specifically for this task.

Simple Text Editor

Using notepad or other editor simply open your settings file.



```
;
; DEFAULT SETTINGS FILE
;



; Change the SHORTCODE value to have different QA scripts built with different names
; Example:
;   ShortCode =               FileName:  QA_v3.16.0000.ps1
;   ShortCode = ACME          FileName:  QA_ACME_v3.16.0000.ps1

[settings]
shortcode         =
language          = en-gb

reportCompanyName = ACME
outputLocation    = $env:SystemDrive\QA\Results\


; To skip a particular check for this configuration, add -SKIP to the end of the check name
; Example:
;   [acc01]                   Use this check
;   [acc01-skip]              Skip this check
;
; ACCOUNTS
[acc01]
IgnoreTheseUsers                      = ('Guest', 'ASPNET', '___VMware')

[acc02]
InvalidAdminNames                     = ('Administrator', 'Admin', 'Guest', 'Guest1')

[acc03]
IgnoreTheseUsers                      = ('Domain Admins', 'Enterprise Admins')

[acc04]
IgnoreTheseUsers                      = ('Allowed RODC Password Replication Group', 'Cert Publishers',
'ConfigMgr Remote Control Users', 'Denied RODC Password Replication Group', 'DHCP', 'DnsAdmins',
'HelpServicesGroup', 'IIS_WPG', 'Offer Remote Assistance Helpers', 'Pre-Windows 2000 Compatible
Access', 'RAS and IAS Servers', 'TelnetClients', 'WinRMRemoteWMIUsers__', 'SQLServer', 'RSABypass')

[acc05]
IgnoreTheseUsers                      = ('NT AUTHORITY\\NetworkService', 'NT AUTHORITY\\LocalService',
'LocalSystem')

[acc06]
; No Settings

[acc07]
AllMustExist                          = 'False'
```

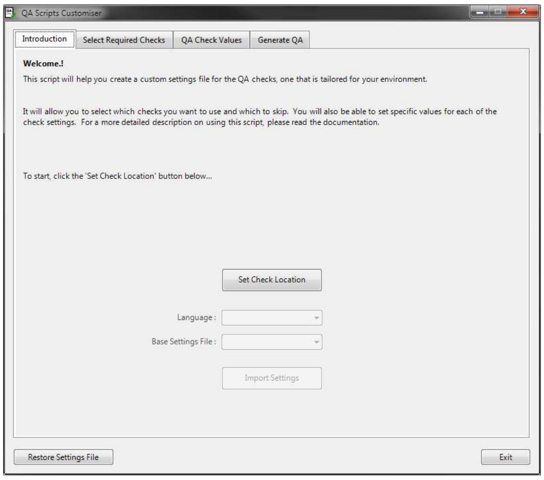
Use this method you need to make a quick change to an existing settings file you have already created. It's not recommended for large scan edits. Once you have completed your edits you will need to recompile the checks into a single QA Script. See **Appendix A** for details.

QA Settings Configurator

The configurator is a PowerShell script that helps you configure a new settings file for your environment. Using Windows Forms, it presents a nice GUI interface that makes it easier to create or edit your settings.

Page: Introduction

To start, execute the `QA-Settings-Configurator.ps1` script in a PowerShell window. After a few seconds, the following window will appear:



Welcome.!

This script will help you create a custom settings file for the QA checks, one that is tailored for your environment.

It will allow you to select which checks you want to use and which to skip. You will also be able to set specific values for each of the check settings. For a more detailed description on using this script, please read the documentation.

To start, click the 'Set Check Location' button below...

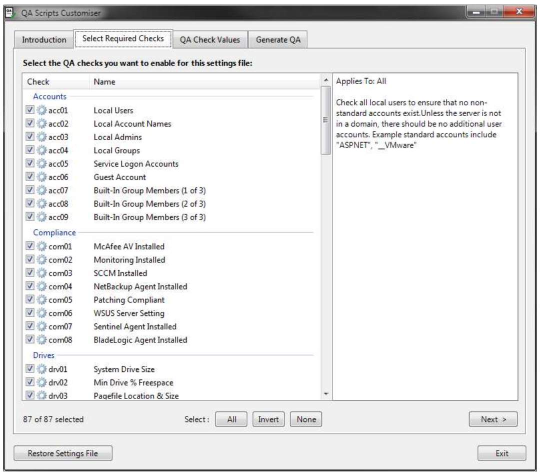
Language :

Base Settings File :

Click the "Set Check Location" button and select the folder where all the checks are located. Next, select the language (currently only English) and a base settings file to use. If you are just starting out, then only the "default-settings" will be available. Click "Import Settings" when ready.

Page: Select Required Checks

The next page lists all the available check and shows which of them are currently enabled. In the default settings file several checks are disabled. If you want to use them, simply tick the box next to the name.



Select the QA checks you want to enable for this settings file:

Check	Name
Accounts	
☑ acc01	Local Users
☑ acc02	Local Account Names
☑ acc03	Local Admins
☑ acc04	Local Groups
☑ acc05	Service Logon Accounts
☑ acc06	Guest Account
☑ acc07	Built-In Group Members (1 of 3)
☑ acc08	Built-In Group Members (2 of 3)
☑ acc09	Built-In Group Members (3 of 3)
Compliance	
☑ com01	McAfee AV Installed
☑ com02	Monitoring Installed
☑ com03	SCCM Installed
☑ com04	NetBackup Agent Installed
☑ com05	Patching Compliant
☑ com06	WSUS Server Setting
☑ com07	Sentinel Agent Installed
☑ com08	BladeLogic Agent Installed
Drives	
☑ drv01	System Drive Size
☑ drv02	Min Drive % Freespace
☑ drv03	Pagefile Location & Size

Applies To: All

Check all local users to ensure that no non-standard accounts exist.Unless the server is not in a domain, there should be no additional user accounts. Example standard accounts include "ASPNET", "__VMware"

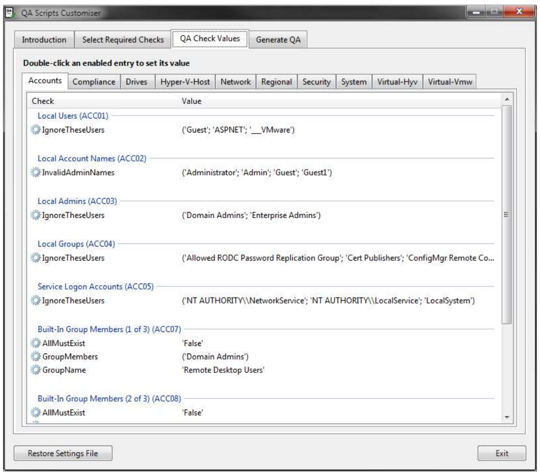
87 of 87 selected

The checks are grouped into sections to help make sense of the larger number of checks. From the screen shot above you can see three of the section: Accounts, Compliance and Drives. The full list is shown later in this document.

Selecting each check will show you some basic information about the check on the right hand side of the window. Once you have selected all the checks you want to include in your settings file, click the "Next >" button.

Page: QA Check Values

This page contains several tabs, one for each of the sections.

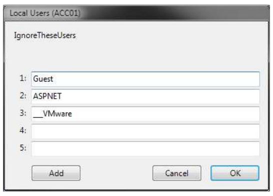


Double-click an enabled entry to set its value

Check	Value
Local Users (ACC01)	
IgnoreTheseUsers	('Guest'; 'ASPNET'; '__VMware')
Local Account Names (ACC02)	
InvalidAdminNames	('Administrator'; 'Admin'; 'Guest'; 'Guest1')
Local Admins (ACC03)	
IgnoreTheseUsers	('Domain Admins'; 'Enterprise Admins')
Local Groups (ACC04)	
IgnoreTheseUsers	('Allowed RODC Password Replication Group'; 'Cert Publishers'; 'ConfigMgr Remote Co...
Service Logon Accounts (ACC05)	
IgnoreTheseUsers	('NT AUTHORITY\\NetworkService'; 'NT AUTHORITY\\LocalService'; 'LocalSystem')
Built-In Group Members (1 of 3) (ACC07)	
AllMustExist	'False'
GroupMembers	('Domain Admins')
GroupName	'Remote Desktop Users'
Built-In Group Members (2 of 3) (ACC08)	
AllMustExist	'False'

Each section tab lists the checks that you have selected as well as the required information for those checks. The default values are already filled in, but you should change this to suit your own environment.

Double-clicking the first item in the screen shot (“IgnoreTheseUsers”) shows us the following window



IgnoreTheseUsers

1: Guest
2: ASPNET
3: __VMware
4:
5:

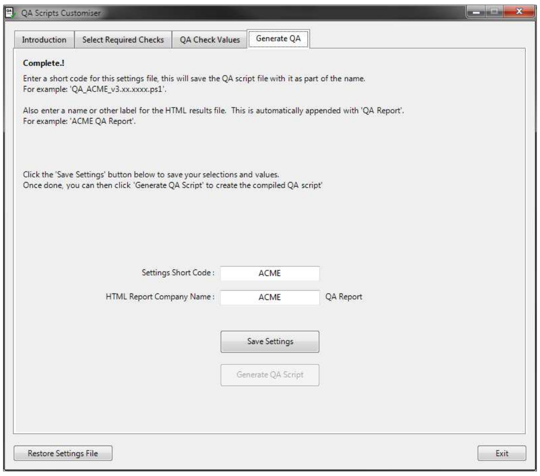
This allows you to change the current values for this check or add new values. In this example, this particular check will see if any local users exist on the server. If there are known domain admin accounts that you want to exclude from this check, this is where you would add them.

Make sure you examine each value for the checks in every section. In most cases the defaults will work for you; however some of the security ones are quite restrictive.

When you are done, click the last page

Page: Generate QA

This last page allows you to set a short code for the settings file you are creating as well as the name that appears as the header on the HTML report



Complete.!

Enter a short code for this settings file, this will save the QA script file with it as part of the name.
For example: 'QA_ACME_v3.xx.xxxx.ps1'.

Also enter a name or other label for the HTML results file. This is automatically appended with 'QA Report'.
For example: 'ACME QA Report'.

Click the 'Save Settings' button below to save your selections and values.
Once done, you can then click 'Generate QA Script' to create the compiled QA script'

Settings Short Code : ACME

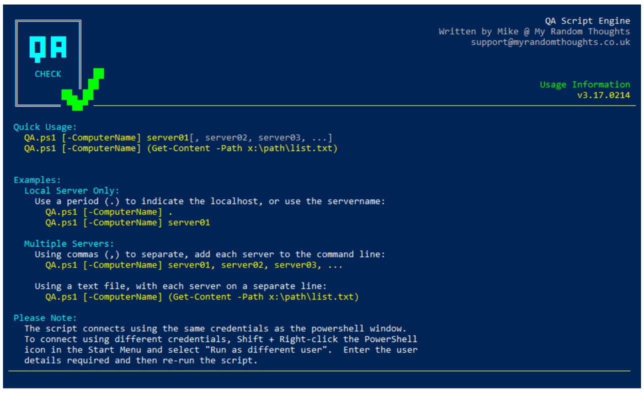
HTML Report Company Name : ACME QA Report

Once you have entered these values, click the “Save Settings” button and give your settings file a name.

Once saved, you can automatically compile your QA script by clicking the “Generate QA Script” button. This will take a few seconds and it will tell you when it’s complete. The generation button performs the same compile function as described in **Appendix A**, and must be done after any settings changes.

Help Screen

When executing the QA script without any command line arguments, the help screen is shown.



```
                                                          QA Script Engine
                                            Written by Mike @ My Random Thoughts
                                               support@myrandomthoughts.co.uk

                                                         Usage Information
                                                               v3.17.0214

Quick Usage:
  QA.ps1 [-ComputerName] server01[, server02, server03, ...]
  QA.ps1 [-ComputerName] (Get-Content -Path x:\path\list.txt)


Examples:
  Local Server Only:
    Use a period (.) to indicate the localhost, or use the servername:
      QA.ps1 [-ComputerName] .
      QA.ps1 [-ComputerName] server01

  Multiple Servers:
    Using commas (,) to separate, add each server to the command line:
      QA.ps1 [-ComputerName] server01, server02, server03, ...

    Using a text file, with each server on a separate line:
      QA.ps1 [-ComputerName] (Get-Content -Path x:\path\list.txt)

Please Note:
  The script connects using the same credentials as the powershell window.
  To connect using different credentials, Shift + Right-click the PowerShell
  icon in the Start Menu and select "Run as different user".  Enter the user
  details required and then re-run the script.
```

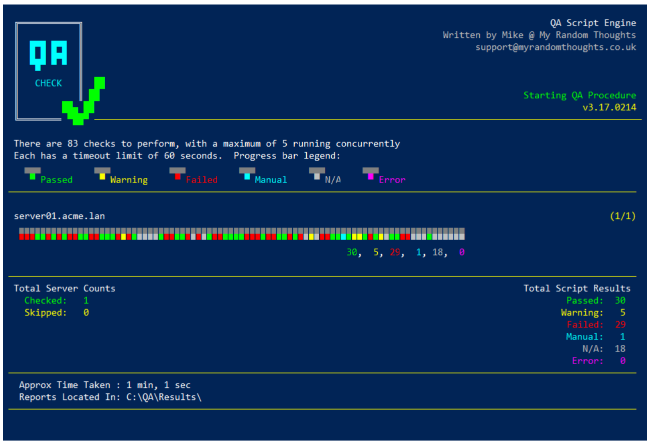
It gives a good overview of the basic command line options available. For a full list of command line options, see **Appendix B**.

Running The Checks…

Against The Local Server

As with the quick start instructions, to run the checks against the local server simply add `localhost` as the computer name. You can also use a full stop (`.`) or the environment variable `$env:ComputerName`

The screen shot below shows a complete scan against my local machine. As you can see it fails on a lot of checks.



```
                                                                  QA Script Engine
                                                Written by Mike @ My Random Thoughts
                                                      support@myrandomthoughts.co.uk
 QA
 CHECK
                                                             Starting QA Procedure
                                                                        v3.17.0214

 There are 83 checks to perform, with a maximum of 5 running concurrently
 Each has a timeout limit of 60 seconds.  Progress bar legend:
    Passed     Warning     Failed     Manual     N/A     Error

 server01.acme.lan                                                             (1/1)

                                                   30,  5, 29,  1, 18,  0

 Total Server Counts                                           Total Script Results
   Checked:   1                                                       Passed:  30
   Skipped:   0                                                      Warning:   5
                                                                      Failed:  29
                                                                      Manual:   1
                                                                         N/A:  18
                                                                       Error:   0

  Approx Time Taken : 1 min, 1 sec
  Reports Located In: C:\QA\Results\
```

During the QA process a coloured progress bar is shown. Each coloured bit is the result of the check that was performed. Note that since checks are performed concurrently, the order of the bits will appear random. The colour coding is as follows:

Green The check **passed** all requirements,
Yellow There was a **warning** for this check,
Red The check **failed** one or more requirements,
Blue This check needs to a **manual** confirmation,
Grey This check is **not applicable** to this server,
Purple There was a critical **error** with this check.

Against Multiple Servers

One of the great features of the QA scripts is that it can check remote servers too. This means you don't need to log on to each server in order to run the scripts.

There are two ways in which you can specify multiple servers. The first is to type them all out on the command line, separated with commas…

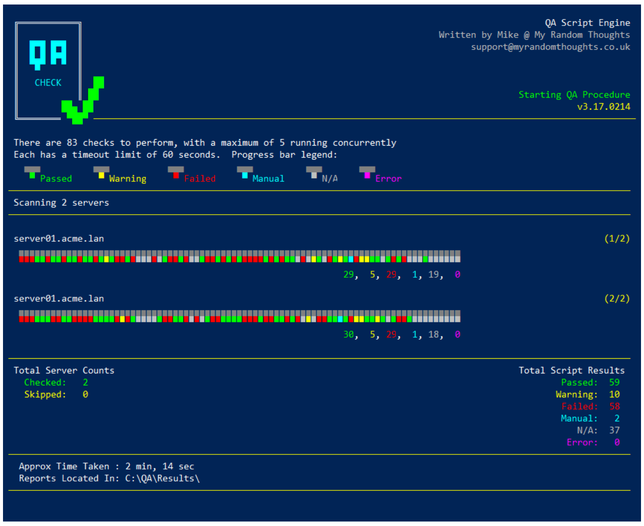
```
.\QA.ps1 -ComputerName server01, server02, server03, server04, …
```

The second way is via a text file with each server listed on its own line…

```
.\QA.ps1 -ComputerName (Get-Content -Path c:\path\file.txt)
```

The `Get-Content` command will read in the file and pass it to the QA scripts.
If you know anything about PowerShell you will see that you can use any command that returns a list of server names in this way. For example, the Active Directory command: `Get-ADComputer`.

When running against more than one server, a different progress bar will appear for each server. Once all servers have been completed the scan results will then be displayed. This is shown in the screen shot below:



```
                                                              QA Script Engine
                                           Written by Mike @ My Random Thoughts
                                                support@myrandomthoughts.co.uk
 CHECK

                                                        Starting QA Procedure
                                                                   v3.17.0214

There are 83 checks to perform, with a maximum of 5 running concurrently
Each has a timeout limit of 60 seconds.  Progress bar legend:
   Passed     Warning     Failed     Manual     N/A     Error

Scanning 2 servers

server01.acme.lan                                                        (1/2)

                                         29,  5, 29,  1, 19,  0

server01.acme.lan                                                        (2/2)

                                         30,  5, 29,  1, 18,  0

Total Server Counts                                        Total Script Results
  Checked:    2                                                    Passed:  59
  Skipped:    0                                                   Warning:  10
                                                                   Failed:  58
                                                                   Manual:   2
                                                                      N/A:  37
                                                                    Error:   0

  Approx Time Taken : 2 min, 14 sec
  Reports Located In: C:\QA\Results\
```

Viewing The Results

While the coloured bars are pretty, they don't give us any information as to which checks passed or failed. If you specified the CSV or XML command lined, there will be different reports you can look it.

The location of all the reports is set to **{system-drive}:\QA\Reports**.

The first is the one that is always generated.

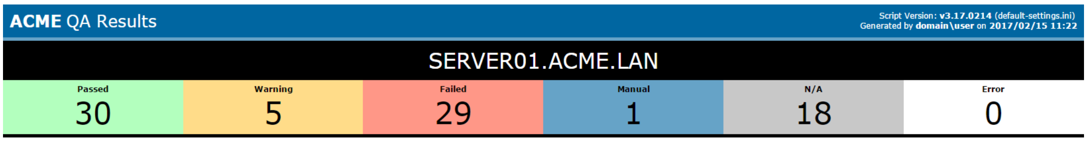
HTML Report

The HTML report give a great visual overview of the results as well as any details on why a particular check returned the result it did. There is also hover help available for each of the checks, giving even more information about the check that was performed.

Header

The HTML header (shown below) includes the information listed which could be used for auditing the build or maintenance processes:

Script version and which settings were used
Who ran this script and when



ACME QA Results

Script Version: **v3.17.0214** (default-settings.ini)
Generated by **domain\user** on **2017/02/15 11:22**

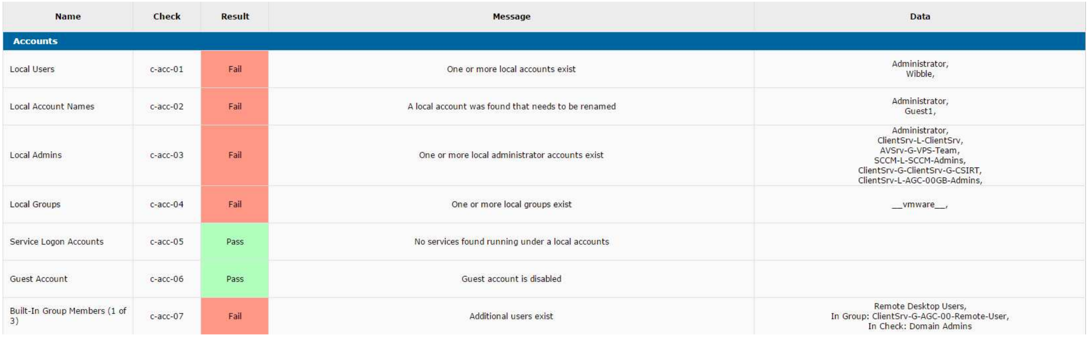
SERVER01.ACME.LAN

Passed	Warning	Failed	Manual	N/A	Error
30	5	29	1	18	0

The header also includes an overview count of the results for this particular server.

Body

The body of the HTML report lists each check grouped in their sections. The screen shot below shows the column headers and the first section (Accounts)

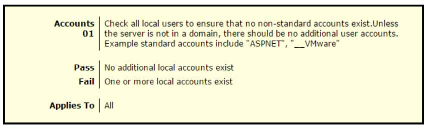


Name	Check	Result	Message	Data
Accounts				
Local Users	c-acc-01	Fail	One or more local accounts exist	Administrator, Wibble,
Local Account Names	c-acc-02	Fail	A local account was found that needs to be renamed	Administrator, Guest1,
Local Admins	c-acc-03	Fail	One or more local administrator accounts exist	Administrator, ClientSrv-L-ClientSrv, AVSrv-G-VPS-Team, SCCM-L-SCCM-Admins, ClientSrv-G-ClientSrv-G-CSIRT, ClientSrv-L-AGC-00GB-Admins,
Local Groups	c-acc-04	Fail	One or more local groups exist	__vmware__,
Service Logon Accounts	c-acc-05	Pass	No services found running under a local accounts	
Guest Account	c-acc-06	Pass	Guest account is disabled	
Built-In Group Members (1 of 3)	c-acc-07	Fail	Additional users exist	Remote Desktop Users, In Group: ClientSrv-G-AGC-00-Remote-User, In Check: Domain Admins

The columns are:

Name	The friendly name of the check,
Check	The internal name of the check (*this is the name I use for reference*),
Result	The final result of the check,
Message	Short message describing the reason for the result value,
Data	Specific data related to the result value.

By default hover help is enabled for all the HTML reports. To view this, move your mouse cursor over the **Results** value for the check. A small window will show appear in the top left corner of the HTML page giving more information about the check. An example of this is shown below:



Accounts 01	Check all local users to ensure that no non-standard accounts exist.Unless the server is not in a domain, there should be no additional user accounts. Example standard accounts include "ASPNET", "__VMware"
Pass	No additional local accounts exist
Fail	One or more local accounts exist
Applies To	All

An optional command line parameter will remove this hover help; simply add the following to the end of your QA command line:

```
-SkipHTMLHelp
```

CSV / XML Reports

There is an option to create a CSV and/or a XML file containing all the details in the HTML report. The difference however is that when scanning multiple servers, only one report file will be generated of each type, containing all the servers and their details.

These can be used as part of an automation process to check the success/fail values for a server.

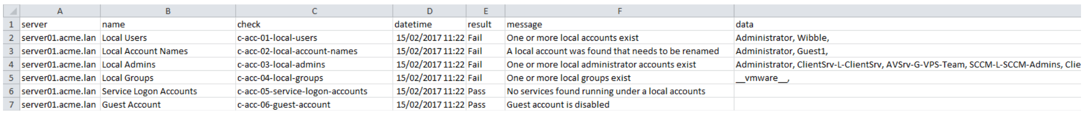
Use the following command line options:

```
-GenerateCSV
-GenerateXML
```

Example Reports

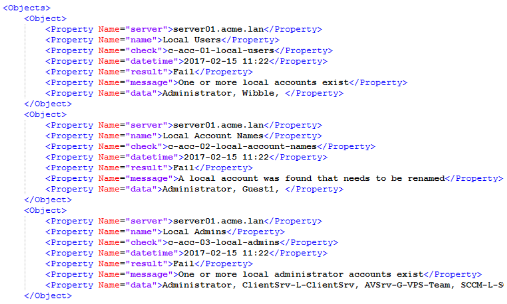
The screen shots below show example data from the generated reports:

CSV Data…



	A	B	C	D	E	F	
1	server	name	check	datetime	result	message	data
2	server01.acme.lan	Local Users	c-acc-01-local-users	15/02/2017 11:22	Fail	One or more local accounts exist	Administrator, Wibble,
3	server01.acme.lan	Local Account Names	c-acc-02-local-account-names	15/02/2017 11:22	Fail	A local account was found that needs to be renamed	Administrator, Guest1,
4	server01.acme.lan	Local Admins	c-acc-03-local-admins	15/02/2017 11:22	Fail	One or more local administrator accounts exist	Administrator, ClientSrv-L-ClientSrv, AVSrv-G-VPS-Team, SCCM-L-SCCM-Admins, Clie
5	server01.acme.lan	Local Groups	c-acc-04-local-groups	15/02/2017 11:22	Fail	One or more local groups exist	__vmware__,
6	server01.acme.lan	Service Logon Accounts	c-acc-05-service-logon-accounts	15/02/2017 11:22	Pass	No services found running under a local accounts	
7	server01.acme.lan	Guest Account	c-acc-06-guest-account	15/02/2017 11:22	Pass	Guest account is disabled	

XML Data…



```
<Objects>
    <Object>
        <Property Name="server">server01.acme.lan</Property>
        <Property Name="name">Local Users</Property>
        <Property Name="check">c-acc-01-local-users</Property>
        <Property Name="datetime">2017-02-15 11:22</Property>
        <Property Name="result">Fail</Property>
        <Property Name="message">One or more local accounts exist</Property>
        <Property Name="data">Administrator, Wibble, </Property>
    </Object>
    <Object>
        <Property Name="server">server01.acme.lan</Property>
        <Property Name="name">Local Account Names</Property>
        <Property Name="check">c-acc-02-local-account-names</Property>
        <Property Name="datetime">2017-02-15 11:22</Property>
        <Property Name="result">Fail</Property>
        <Property Name="message">A local account was found that needs to be renamed</Property>
        <Property Name="data">Administrator, Guest1, </Property>
    </Object>
    <Object>
        <Property Name="server">server01.acme.lan</Property>
        <Property Name="name">Local Admins</Property>
        <Property Name="check">c-acc-03-local-admins</Property>
        <Property Name="datetime">2017-02-15 11:22</Property>
        <Property Name="result">Fail</Property>
        <Property Name="message">One or more local administrator accounts exist</Property>
        <Property Name="data">Administrator, ClientSrv-L-ClientSrv, AVSrv-G-VPS-Team, SCCM-L-S
    </Object>
```

Appendix A – Recompiling the QA scripts

Whenever a change is made to the settings file or any of the individual checks, you will need to recompile the QA script.

The reason for compiling into a single file is to make the completed script portable without having 100 separate files all over, potentially being of different versions.

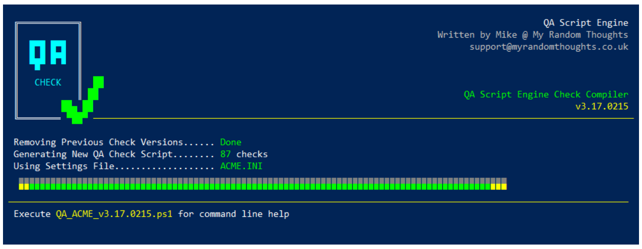
To start, open a normal PowerShell window and change to the folder containing all the script files. Typing the following command will compile all the scripts using the default settings file:

```
.\compiler.ps1
```

To tell the compile script to use one of your settings files type:

```
.\comiler.ps1 -Settings {name_of_file}.ini
```

The screen shot below shows the completed process for a settings file called ACME.INI:



```
                                                    QA Script Engine
                                   Written by Mike @ My Random Thoughts
                                        support@myrandomthoughts.co.uk

                                      QA Script Engine Check Compiler
                                                          v3.17.0215

Removing Previous Check Versions...... Done
Generating New QA Check Script....... 87 checks
Using Settings File.................. ACME.INI

Execute QA_ACME_v3.17.0215.ps1 for command line help
```

As you can see, the QA script filename contains the short code that is used for this settings file.

Appendix B - Command Line Options

There are several command line options that can be used with the QA scripts. These are detailed below:

```
.\QA.ps1
```

-ComputerName	Allows you to add one or more servers to scan,
-SkipHTMLHelp	Removes the hover help from the HTML report,
-GenerateCSV	Outputs a single results.csv file for all servers scanned,
-GenerateXML	Outputs a single results.xml file for all servers scanned,
-Help	Shows the help screen,
-Verbose	Performs one check at a time, for use when debugging checks.

Appendix C – List Of Check And Their Sections

Accounts

These checks are for local user and group configurations

acc01	Local Users
acc02	Local Account Names
acc03	Local Admins
acc04	Local Groups
acc05	Service Logon Accounts
acc06	Guest Account
acc07	Built-In Group Members (1 of 3)
acc08	Built-In Group Members (2 of 3)
acc09	Built-In Group Members (3 of 3)

Compliance

These checks are to make sure the servers meet your tooling compliance targets

com01	McAfee Antivirus Installed
com02	SCOM Monitoring Installed
com03	SCCM Installed
com04	NetBackup Agent Installed
com05	Last Patch Date
com06	WSUS Server
com07	Sentinel Agent Installed
com08	BladeLogic Agent Installed

Drives

These are drive and storage related checks

drv01	System Drive Space
drv02	Minimum Drive Free Space
drv03	Page File Size And Location
drv04	CD/DVD Drive
drv05	Shared Folders
drv06	SAN Storage
drv07	Disk Management Agent
drv08	Drive NTFS Format

Hyper-V Host

These checks are for Microsoft Hyper-V host servers. For Hyper-V guest servers, see the **Virtual-HYV** section.

hvh01	Server Core Edition
hvh02	No Other Server Roles
hvh03	VM Storage Location
hvh04	Integration Services

Network

These checks are for all network related functions

net01	No IPv6
net02	Unused Network Interfaces
net03	Network Adapter Labels
net04	Binding Order
net05	Network Speed And Duplex
net06	Network Agent
net07	Network Teaming
net08	Management Adapter
net09	Static Routes
net10	Power Management
net11	DNS Settings

Regional

These checks are to make sure the servers are in the correct region

reg01	Local Time Setting
reg02	Time Zone Setting
reg03	Location
reg04	Language

Security

These are all security related and are quite strict by default

sec01	SChannel 1 - Ciphers
sec02	SChannel 2 - Hashes
sec03	SChannel 3 - Key Exchange Algorithms
sec04	SChannel 4 - Protocols
sec05	SChannel 5 - Cipher Order
sec06	Reject Enumerate Accounts
sec07	Reject Enumerate Shares
sec08	Domain Credential Caching
sec09	Request Admin Elevated Prompt
sec10	Anonymous Pipe Share Access
sec11	IIS Default Page
sec12	SMB Signing On
sec13	RSA Authentication
sec14	Firewall Rules
sec15	Firewall State
sec16	Open Ports

System

These checks are all system related

sys01	Pending Reboot
sys02	Windows Licence
sys03	Services Not Started
sys04	Services Not Stopped
sys05	System Event Log
sys06	Application Event Log
sys07	Devices Status
sys08	(*does not currently exist*)
sys09	Scheduled Tasks
sys10	Print Spooler
sys11	Auto-Run Disabled
sys12	SNMP Configuration
sys13	Domain Member
sys14	Power Plan Setting
sys15	Hibernation Settings
sys16	Remote Desktop Settings
sys17	Terminal Services Licenced
sys18	Current OU Location

Virtual-HYV

These checks are for Microsoft Hyper-V guest servers. For Hyper-V host servers, see the **Hyper-V-Host** section.

vhv01	Integration Services Version

Virtual-VMW

These checks are for VMware ESX guest servers.

vmw01	Tools Version
vmw02	Time Sync Setting
vmw03	NIC Type
vmw04	LSI SAS Controller
vmw05	SCSI Drive Count
vmw06	Total VM Size
vmw07	CD/DVD/Floppy Drive Mounted

vmw08	Check vOSOT Registry (*experimental*)
vmw09	Check vOSOT Services And Tasks (*experimental*)