Evaluate-STIG Guide & Documentation

What is Evaluate-STIG?

- Evaluate-STIG acts as a sort of SCAP Scan 2.0
- Evaluate-STIG is a PowerShell Script that works with its own library of supported STIG Checklists
- The user can choose to target one, multiple, or even all STIG Checklists at once
- Evaluate-STIG will look at everything on the target machine, from Group Policy Values, to Registry Values, and even more
- After it is done running the provided STIG Checklists you set it to target, it will output a .CKL file, which can be opened in STIG Viewer
    - This .CKL file will also be the same as if you ran a SCAP Scan for the specific checklist, and also imported the .XCCDF data into the checklist
        - Yet, Evaluate-STIG will look at, and validate, way more than a traditional SCAP Scan
- You can also give it preset outputs for various different checklist items in an "AnswerFile"
    - This allows you have it auto-fill a response in the "Finding Details", for if something should have been found, if it's still open, or should not have been found
    - This can save a lot of time when it comes to filling out responses for checklist items that you know will fail, even though they have been addressed.

How to Get Evaluate-STIG

- **[REDACTED For Security Purposes]**

Simplified Step-by-Step Guide (Without Use of an AnswerFile)

This guide assumes you have already imported the latest DISA STIG GPOs, and that you have STIG Viewer 2.17 already installed on the target device. This guide will focus on how to use Evaluate-STIG, without the use of an "AnswerFile".

1. Copy the unzipped "Evaluate-STIG" Folder onto your Target Device, in the Directory of your choice.
2. Run PowerShell as Administrator
3. Parse to the "Evaluate-STIG" folder location in PowerShell: Type in "**cd [Directory Path]**" and then hit "**Enter**"
    a. Ex: "**cd C:\Users\Guest\Evaluate-STIG**"
4. **IMPORTANT:** This next change is temporary, and must be switched back to the way it was after the scan has completed.
5. Type in "**Get-ExecutionPolicy**", and press "**Enter**"
    a. **IMPORTANT:** Write down the output of this command, it will be adjusted accordingly later.
6. Type in "**Set-ExecutionPolicy RemoteSigned**", and press "**Enter**"
    a. Type in a capital "**A**", and then press "**Enter**"
7. For a Simplified Scan type in "**.\Evaluate-STIG.ps1 -ScanType [Unclassified or Classified] -OutputPath [The directory path for the output] -Output [CKL, CombinedCLK, STIGManager,... etc.] -SelectSTIG [Win10,MSEdge,... etc.]**"
    a. Example Input: "**.\Evaluate-STIG.ps1 -ScanType Classified -OutputPath "C:\Admin\3.5. Evaluate-STIG" -Output CKL -SelectSTIG Win10**"
    b. For the **-SelectSTIG** option, if you wanted to do more than one STIG: "**-SelectSTIG** Win10,MSEdge"
    c. **Note:** There is a lot of different command combinations, and supported STIGs for this tool. This is just one example of a combination of options for the command.
8. There should be a Red Bar at the top of the PowerShell Window, telling you the progress of the scan. Once it has completed, it will say:
    a. "**Applicable STIGs to process - #**"
    b. "**Done!**"
    c. "**Total Time : (time)**"
    d. "**Total CKLs in Results Directory : 2**"
    e. "**Results saved to C:\OutputPath provided beforehand**"
9. **IMPORTANT:** Type in "**Set-ExecutionPolicy Restricted**", and press "**Enter**"
    a. "**Restricted**" would be replaced by the output for "**Get-ExecutionPolicy**" that you wrote-down before.
    b. Type in a capital "**A**", and then press "**Enter**"
10. Open STIG Viewer as Administrator
11. Select the "STIG Explorer" Tab, then Click on "ChecklistOpen Checklist from File".
    a. Parse to your specified Evaluate-STIG "**OutputPath**" destination.
    b. Open the Folders "COMPUTERNAME", "Checklist", and then Select the Checklist (.CKL) File, and click "Open"
12. Now you're good to go! 🙂

More details and specifics can be found in the "**/Evaluate-STIG_1.2307.2.zip/doc/Evaluate-STIG_UserGuide-1.2307.0.pdf**" file.

Useful Pages

- PowerShell (Command) Usage
    - **Pgs 8-12**
- AnswerFile Configuration
    - **Pgs 12-15**

What are Evaluate-STIG AnswerFiles?

- Evaluate-STIG is already really cool, but AnswerFiles take Evaluate-STIG to the next level.
- Were you ever hardening multiple of the same machine, with all of the same consistent changes made on all of them?
- Did you find that SCAP Compliance Checker didn't pick up on the same GPOs that were already hardened, and instead just marked them as NR?
- If this is the case, AnswerFiles are just the thing for you!
- An AnswerFile is a sort of Answer Key, or drop down answer box, for a STIG Checklist.
- You can use AnswerFiles to auto-fill your desired comments, and change a given checklist item to the appropriate status.
    - This is extremely useful, and should only be used, if you know for a fact that the checklist item has already been hardened.

Evaluate-STIG Information & Guide on AnswerFiles - Table of Contents

Important Guidelines for Making an Evaluate-STIG AnswerFile

- When utilizing AnswerFile(s), it is best practice to keep a separate folder for your library of AnswerFiles outside of the directory of Evaluate-STIG.
- Only keep one AnswerFile in the **~\Evaluate-STIG\AnswerFiles** directory at a time.
    - If there are multiple AnswerFiles in the **~\Evaluate-STIG\AnswerFiles** directory, it will typically go with the one that gives it the best results, not always the correct one for a given use-case.
- You should have one AnswerFile for a given STIG Checklist.
    - Ex: **MSEdge_AnswerFile.xml** and **Win_10_AnswerFile.xml**
    - This will not only keep your AnswerFiles more organized, but it will also enable you to detect any issues with the given files, if there is one.
        - Ex: If you have a Syntax Error or Formatting Error in a given AnswerFile

How to Make a Basic Evaluate-STIG AnswerFile

The Evaluate-STIG Folder comes with a great template for creating AnswerFiles. This template can be found in **~\Evaluate-STIG\AnswerFiles\Template_AnswerFile.xml**

- **Tip:** Visual Studio Code makes it way easier to see, edit, and understand the contents of an .XML file

Template_AnswerFile.xml

```
<?xml version="1.0" encoding="utf-8"?>
<!--**************************************************************************************
This file contains answers for known opens and findings that cannot be evaluated through technical means.
<STIGComments Name> must match the STIG ShortName or Name in -ListSupportedProducts.  When a match is found,
this answer file will automatically for the STIG.
<Vuln ID> is the STIG VulnID.  Multiple Vuln ID sections may be specified in a single Answer File.
<AnswerKey Name> is the name of the key assigned to the answer.  "DEFAULT" can be used to apply the comment to
any asset.  Multiple AnswerKey Name sections may be configured within a single Vuln ID section.
<ExpectedStatus> is the initial status after the checklist is created.  Valid entries are "Not_Reviewed",
"Open", "NotAFinding", and "Not_Applicable".
<ValidationCode> must be Powershell code that returns a True/False value.  If blank, "true" is assumed.
<ValidTrueStatus> is the status the check should be set to if ValidationCode returns "true".  Valid entries are
"Not_Reviewed", "Open", "NotAFinding", and "Not_Applicable".  If blank, <ExpectedStatus> is assumed.
<ValidTrueComment> is the verbiage to add to the Comments section if ValidationCode returns "true".
<ValidFalseStatus> is the status the check should be set to if ValidationCode DOES NOT return "true".  Valid
entries are "Not_Reviewed", "Open", "NotAFinding", and "Not_Applicable".  If blank, <ExpectedStatus> is assumed.
<ValidFalseComment> is the verbiage to add to the Comments section if ValidationCode DOES NOT return "true".
**************************************************************************************-->
<STIGComments Name="">
  <Vuln ID="">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus></ValidTrueStatus>
      <ValidTrueComment></ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
</STIGComments>
```

1. First, Create a New Folder Outside your Evaluate-STIG Directory (This will be the location for your Library of AnswerFiles).
2. Next, navigate to where you have stored your Evaluate-STIG Folder, and then navigate to the "**AnswerFiles**" Folder from there.
    a. Ex: **~\Evaluate-STIG\AnswerFiles**
3. Inside this folder there will be a File called "**Template_AnswerFile.xml**". Copy this File, and Paste it into your Newly-Created Folder for your AnswerFile Library.
4. Once you have done this, Rename the Copied File to whatever you like.
    a. In this Example I will be using the File Name: "**MSEdge_AnswerFile_1V-ID"** (with the File Type being a .XML File)
5. Now, let's Open this Newly Created File. You can use Visual Studio Code, Notepad, or any other Application of your choice that can Open and Modifiy .XML Files.
6. First, Type in the STIG Checklist Name that you want to Target in the Quotes for: **<STIGComments Name="">**
    a. Ex: **<STIGComments Name="Microsoft Edge">**
7. Next, Type in the Vulnerability ID for the STIG Checklist Item that you want to Target in the Quotes for: **<Vuln ID="">**
    a. Ex: **<Vuln ID="V-235758">**
8. Type in the AnswerKey Name that you want to Target in the Quotes for: **<AnswerKey Name="">**
    a. Ex: **<AnswerKey Name="DEFAULT">**
9. Type in the Status that you Expect Evaluate-STIG to Result in for this Checklist Item (Ex: **Not_Reviewed**, **Not_Applicable**, **NotAFinding**, **Open**): **ExpectedStatus>Not_Reviewed**</**ExpectedStatus>**
    a. Ex: **<ExpectedStatus>Not_Reviewed**</**ExpectedStatus>**
10. Type in the Status that you want Evaluate-STIG to Change this Vulnerability ID's Output of: **<ValidTrueStatus>**</**ValidTrueStatus>**
    a. Ex: **<ValidTrueStatus>NotAFinding**</**ValidTrueStatus>**
11. Type in what Comment you want Evaluate-STIG to Input for this Vulnerability ID in the STIG Checklist: **<ValidTrueComment>**</**ValidTrueComment>**
    a. Ex: **<ValidTrueComment>NF - The latest version of Edge is Installed. -TK 3/29/2023**</**ValidTrueComment>**
12. Save the .XML File that you have created.
13. Now move the "**Template_AnswerFile.xml**" from the **~\Evaluate-STIG\AnswerFiles** Folder, into your previously created AnswerFile Library.
14. Now move your newly created AnswerFile into the **~\Evaluate-STIG\AnswerFiles** Folder.
    a. **Note:** There should only be one File in your **~\Evaluate-STIG\AnswerFiles** Folder, and this should be the one you just created.

Example of Basic Evaluate-STIG AnswerFile

MSEdge_AnswerFile_1V-ID.xml

```
<?xml version="1.0" encoding="utf-8"?>
<STIGComments Name="Microsoft Edge">
  <Vuln ID="V-235758">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - The latest version of Edge is Installed. -TK 3/29/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
</STIGComments>
```

How to Use an Evaluate-STIG AnswerFile

1. Now that you have created your own AnswerFile, let's put it to use!
2. The process of using an AnswerFile is almost identical as using Evaluate-STIG without an AnswerFile, so these steps might seem a bit familiar.
3. Run PowerShell as Administrator
4. Parse to the "Evaluate-STIG" folder location in PowerShell: Type in "**cd [Directory Path]**" and then hit "**Enter**"
    a. Ex: "**cd C:\Users\Guest\Evaluate-STIG**"
5. **IMPORTANT:** This next change is temporary, and must be switched back to the way it was after the scan has completed.
6. Type in "**Get-ExecutionPolicy**", and press "**Enter**"
    a. **IMPORTANT:** Write down the output of this command, it will be adjusted accordingly later.
7. Type in "**Set-ExecutionPolicy RemoteSigned**", and press "**Enter**"
    a. Type in a capital "**A**", and then press "**Enter**"
8. For a Simplified Scan type in "**.\Evaluate-STIG.ps1 -ScanType [Unclassified or Classified] -AnswerKey [DEFAULT or your Custom STIGComments Name] -OutputPath [The directory path for the output] -Output [CKL, CombinedCLK, STIGManager,... etc.] -SelectSTIG [Win10,MSEdge,... etc.]**"
    a. Example Input: "**.\Evaluate-STIG.ps1 -ScanType Classified -AnswerKey DEFAULT -OutputPath "C:\Admin\3.5. Evaluate-STIG" -Output CKL -SelectSTIG MSEdge**"
9. There should be a Red Bar at the top of the PowerShell Window, telling you the progress of the scan. Once it has completed, it will say:
    a. "**Applicable STIGs to process - #**"
    b. "**Done!**"
    c. "**Total Time : (time)**"
    d. "**Total CKLs in Results Directory : 1**"
    e. "**Results saved to C:\OutputPath provided beforehand**"
10. **IMPORTANT:** Type in "**Set-ExecutionPolicy Restricted**", and press "**Enter**"
    a. "**Restricted**" would be replaced by the output for "**Get-ExecutionPolicy**" that you wrote-down before.
    b. Type in a capital "**A**", and then press "**Enter**"
11. Open STIG Viewer as Administrator
12. Select the "STIG Explorer" Tab, then Click on "ChecklistOpen Checklist from File".
    a. Parse to your specified Evaluate-STIG "**OutputPath**" destination.
    b. Open the Folders "COMPUTERNAME", "Checklist", and then Select the Checklist (.CKL) File, and click "Open"
13. Now your Evaluate-STIG AnswerFile should have been utilized, and you're good to go! 🙂
    a. You can also verify if the AnswerFile has in-fact worked, by Selecting the Vulnerability ID that you were Targeting, and by looking at the Comments Section.

Evaluate-STIG AnswerFile Word Bank

Here is a break-down of all the Components Associated with an AnswerFile.

1. **<STIGComments Name="Microsoft Edge">**
    a. This Section refers to the Checklist that it will be Targeting.
    b. For example, if I wanted to make an AnswerFile for the **"Microsoft Edge"** STIG Checklist, I would put **"Microsoft Edge"** in the Quotes, or I could also put the Approved Short-Name in the Quotes as well.
        i. Example with Approved Short-Name: **<STIGComments Name="MSEdge">**
2. **<Vuln ID="V-235758">**
    a. This Section refers to the Specific **"Vulnerability ID"** that the Answer Key will be Targeting, as it appears in the latest STIG Checklist.
3. **<AnswerKey Name="DEFAULT">**
    a. This Section refers to the Mode, Profile, or Setting you want a Specific Answer Key Response (Vulnerability-ID) to Associate with.
    b. When Run, Unless Specified, Evaluate-STIG will always go with the **"DEFAULT"** Answer Key Response, if one exists.
    c. **IMPORTANT: AnswerKey Name="DEFAULT"** must be present, in order for the AnswerFile to work. Any additional **AnswerKey Name**s are fine, as long as there is one **"DEFAULT"** per Vulnerability ID.
4. <**ExpectedStatus**>**Not_Reviewed**</**ExpectedStatus**>
    a. This Section refers to what you would Expect Evaluate-STIG to Output After the Scan has Completed.

    b. For example, if I knew that V-235758 in the Microsoft Edge STIG was in-fact completed ("Fix Text: Install a supported version of Edge."), and I still expected Evaluate-STIG to leave it as **Not_Reviewed**, I would specify **Not_Reviewed** in **ExpectedStatus**

5. **Supported Statuses - Not_Reviewed**, **NotAFinding**, **Open**, **Not_Applicable**
    a. These are all the Currently-Supported Statuses that you could Utilize for a given **Status** Field.
6. <**ValidTrueStatus>NotAFinding**</**ValidTrueStatus>**
    a. This Section refers to what you would want Evaluate-STIG to Overwrite this Checklist item as, if your previously defined **ExpectedStatus** was True.
    b. For example, if I knew that V-235758 in the Microsoft Edge STIG was in-fact completed ("Fix Text: Install a supported version of Edge."), and I still expected Evaluate-STIG to leave it as **Not_Reviewed**, I would specify **NotAFinding** in **ValidTrueStatus**, since this Checklist Item has already been completed.
7. <**ValidTrueComment>NF - The latest version of Edge is Installed. -TK 3/29/2023**</**ValidTrueComment>**
    a. This Section refers to what you would want specified in the Comments Section for the Given Vulnerability ID, in the STIG Checklist, if the **ExpectedStatus** was True.
8. **<ValidFalseStatus>Not_Reviewed</ValidFalseStatus>**
    a. This Section refers to what you would want the Checklist ID Status to be, if the Evaluate-STIG Result is not what you had Expected.
    b. For example, if I thought that V-235758 in the Microsoft Edge STIG was going to be flagged as **Not_Reviewed**, but it actually turned out to be **NotAFinding**, then this **ValidFalseStatus** would set this Checklist Item back to **Not_Reviewed**, so I could manually look into the STIG Checklist Item.
9. **<ValidFalseComment>NR - This Checklist Item needs to be looked at. -TK 3/29/2023</ValidFalseComment>**
    a. This Section refers to what you would want specified in the Comments Section for the Given Vulnerability ID, if the **ExpectedStatus** was NOT True.

How to Make an Evaluate-STIG AnswerFile with 2 Vulnerability-IDs

Having an AnswerFile that Targets One Vulnerability-ID is nice, but what if there were Multiple Vulnerability-IDs that we wanted to take care of all at once?

- The **Microsoft Edge** Evaluate-STIG Scan will always leave "**V-235758**" as "**Not_Reviewed**", and "**V-235719**" as "**Open**", even if these STIG Checklist Items have already been addressed.
- If you wanted to Create an AnswerFile that contained Two or More Vulnerability-IDs, you could Copy the Section of the AnswerFile From "**<Vuln ID ="V-#">**" to "**</Vuln>**", and Paste it Below the Last Instance of "**</Vuln>**".
- An Example of an AnswerFile with 2 Vulnerability-IDs can be Seen Below.

Example of Evaluate-STIG AnswerFile with 2 Vulnerability-IDs

MSEdge_AnswerFile_2V-ID.xml

```
<?xml version="1.0" encoding="utf-8"?>
<STIGComments Name="Microsoft Edge">
  <Vuln ID="V-235758">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - The latest version of Edge is Installed. -TK 3/29/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-235719">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Open</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - The "Proxy Settings" GPO was already set to "Enabled", with "direct" for the Proxy
Settings. -TK 3/29/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
</STIGComments>
```

How to Make an Evaluate-STIG AnswerFile with 3 AnswerKey Names (3 Profiles)

What if we wanted to make One AnswerFile for an Application, or Service, that is Used on Multiple Different Operating Systems?

- This is an Extreme Example, and Edge wouldn't be Installed on Linux, but say we wanted to Create an AnswerFile for the **Microsoft Edge** STIG Checklist. But Instead of Making Multiple AnswerFiles for a Device that has the Latest Version of Edge Installed: "**NotAFinding**", a Device that wouldn't need to have Edge Installed: "**Not_Applicable**", and another Device that has an Outdated Operating System, which still needs to be upgraded: "**Open**", we could Address All of these Scenarios in One AnswerFile.
- To Create Additional Options for an AnswerFile, Copy the Section of the AnswerFile From "**<AnswerKey Name="">**" to "<**/AnswerKey>**", and Paste it Below the Last Instance of "<**/AnswerKey>**".
- An Example of an AnswerFile with 1 Vulnerability-ID and 3 Options can be Seen Below.

Example of Evaluate-STIG AnswerFile with 3 AnswerKey Names (3 Profiles)

MSEdge_AnswerFile_1V-ID-3Opt.xml

```
<?xml version="1.0" encoding="utf-8"?>
<STIGComments Name="Microsoft Edge">
  <Vuln ID="V-235758">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - The latest version of Edge is Installed. -TK 3/29/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="Linux">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Not_Applicable</ValidTrueStatus>
      <ValidTrueComment>N/A - Edge won't be Installed on Linux. -TK 4/27/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="Outdated_OS">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Open</ValidTrueStatus>
      <ValidTrueComment>Open - The Latest Version of Edge still needs to be Installed. -TK 4/27/2023<
/ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
</STIGComments>
```

How to Make an Evaluate-STIG AnswerFile with 10 Vulnerability-IDs

What if we wanted to Combine what we Learned from the Previous Two Sections, and Create a Proper Evaluate-STIG AnswerFile to Address all of the "**Open**" and "**Not_Reviewed**" Items in a given STIG Checklist?

- To Create an Evaluate-STIG AnswerFile for a given STIG Checklist, such as the **Windows 10** Checklist, you can Simply Combine the Knowledge Learned from the Two Sections Above, "**How to Make an Evaluate-STIG AnswerFile with 2 Vulnerability-IDs**" and "**How to Make an Evaluate-STIG AnswerFile with 3 AnswerKey Names (3 Profiles)**".
- **For Each Vulnerability-ID that you want to Target:** Copy the Section of the AnswerFile From "**<Vuln ID="V-#">**" to "**</Vuln>**", and Paste it Below the Last Instance of "**</Vuln>**".
    - Make sure you Pay Attention to the **Vulnerability-ID Numbers** for Each Vulnerability
    - Make sure you Match the <**ExpectedStatus**> to the Correct Status that Appears for that Vulnerability-ID
    - Make sure you Match the <**ValidTrueStatus**> to the Correct Status that Applies to the Device and Vulnerability-ID that you are Targeting
    - Make sure the <**ValidTrueComment**> is Unique, and Applies to the Device and Vulnerability-ID that you are Targeting
    - Feel Free to Add Contents for <**ValidFalseStatus**> and <**ValidFalseComment**> Wherever you see fit
- If you wanted to Address Multiple Devices with a Single AnswerFile, you Could Follow this Guide, and then Add Option(s), "**Answer Key Name**"(s) , to the AnswerFile, According to the "**How to Make an Evaluate-STIG AnswerFile with 3 AnswerKey Names (3 Profiles)**" Guide Above.
- An Example of an AnswerFile with 10 Vulnerability-IDs can be Seen Below.

Example of Evaluate-STIG AnswerFile with 10 Vulnerability-IDs (With Flags for Open, NotAFinding, Not_Applicable, and Not_Reviewed)

Windows_10_AnswerFile_10V-ID_Proposal_Laptops.xml

```
<?xml version="1.0" encoding="utf-8"?>
<STIGComments Name="Win10">
  <Vuln ID="V-220702">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Open</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Open</ValidTrueStatus>
      <ValidTrueComment>Open - Bitlocker will be configured at a later date. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220717">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Open</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Open</ValidTrueStatus>
      <ValidTrueComment>Open - This still needs to be configured. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220921">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Open</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - "Interactive logon: Message text for users attempting to log on" was already
configured. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220922">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Open</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - "Interactive logon: Message title for users attempting to log on" was already
configured. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220701">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Open</ValidTrueStatus>
      <ValidTrueComment>Open - Up to ISSO/ISSM iterpretation. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220712">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - Only Authorized Administrators are in the Administrator Group. -TK 4/4/2023<
/ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220725">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Not_Applicable</ValidTrueStatus>
      <ValidTrueComment>N/A - This is Not Applicable for Standalone Workstations. -TK 4/4/2023<
/ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220737">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>NotAFinding</ValidTrueStatus>
      <ValidTrueComment>NF - Firewall policies are set to deny all, with specific in-bound and out-bound rules
to allow ports and services. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220738">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Not_Applicable</ValidTrueStatus>
      <ValidTrueComment>N/A - VMs do not exist on this Workstation. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
  <Vuln ID="V-220903">
    <!--Optionally enter STIG rule title as XML comment for easy reference-->
    <AnswerKey Name="DEFAULT">
      <ExpectedStatus>Not_Reviewed</ExpectedStatus>
      <ValidationCode></ValidationCode>
      <ValidTrueStatus>Not_Applicable</ValidTrueStatus>
      <ValidTrueComment>N/A - We're not using DoD Root CA on our Systems. -TK 4/4/2023</ValidTrueComment>
      <ValidFalseStatus></ValidFalseStatus>
      <ValidFalseComment></ValidFalseComment>
    </AnswerKey>
  </Vuln>
</STIGComments>
```

More details and specifics can be found in the "**/Evaluate-STIG_1.2307.2.zip/doc/Evaluate-STIG_UserGuide-1.2307.0.pdf**" file.

Useful Pages

- PowerShell (Command) Usage
  - **Pgs 8-12**
- AnswerFile Configuration
  - **Pgs 12-15**

----------------------------WORK IN PROGRESS----------------------------

How does Remote Scanning Work with Evaluate-STIG?

- Evaluate-STIG can be utilized to scan multiple devices at once, from one "host" device.
- The functionality of remote scanning in Evaluate-STIG works almost identical as SCAP Compliance Checker, with the added benefits of Evaluate-STIG's enhanced capabilities.
- When it comes to running remote Evaluate-STIG scans, the user can choose to target one, multiple, or even all STIG Checklists at once.
  - Note: The STIG Checklists that you specify in these multiple-device Evaluate-STIG scans, will all be executed on every device.

        - Example: If you specified "Win10,MSEdge,AdobeAcrobatProDCContinuous", and one of the devices wasn't running Windows 10, or didn't have Adobe Acrobat Reader DC Continuous, those STIG checklists would still be run on those machines.
- Evaluate-STIG will look at everything on the target machine, from Group Policy Values, to Registry Values, and even more, on every targeted machine.
- After it is done running the provided STIG Checklists you set it to target, it will output .CKL file(s), for each STIG Checklist corresponding with each device, which can be opened in STIG Viewer.
    - These .CKL file will also be the same as if you ran a SCAP Scan for the specific checklist, and also imported the .XCCDF data into the checklist.
        - Yet, Evaluate-STIG will look at, and validate, way more than a traditional SCAP Scan would.
- You can also give it preset outputs for various different checklist items in an "AnswerFile".
    - This allows you have it auto-fill a response in the "Finding Details", for if something should have been found, if it's still open, or should not have been found.
    - This can save a lot of time when it comes to filling out responses for checklist items that you know will fail, even though they have been addressed.
    - When it comes to remote scanning with Evaluate-STIG, specified AnswerFiles will be used across all devices for a given remote scan.
        - As long as you configure the different components of the AnswerFiles correctly, this shouldn't be an issue, but it is worth knowing.

----------------------------WORK IN PROGRESS----------------------------

Evaluate-STIG Remote Scanning Prerequisites

For more technical details and Microsoft docs/guides, see the User Guide sections 2.2 and 3.7

- Make sure you have Evaluate-STIG on the host device that will be remotely-scanning the other client devices.
- Make sure the host and client devices are on the same network.
    - Make sure you can ping from the host device, to all the client devices.
    - Make sure you can ping from the client devices, to the host device.
- Ensure at least one of the following (primary bullet points below) is configured
    - The devices are a part of the same Domain and WorkGroup.
    - The devices are added in the host device's "hosts" file, and an exception is made in WSMan (WS-Management) (Windows only).
        - This is so that the target devices can be resolvable through DNS.
        - "hosts" File Location: "C:\Windows\System32\drivers\etc\hosts"

----------------------------WORK IN PROGRESS----------------------------

Simplified Step-by-Step Guide (Without Use of an AnswerFile)

This guide assumes you have already imported the latest DISA STIG GPOs, and that you have STIG Viewer 2.17 already installed on the target device. This guide will focus on how to use Evaluate-STIG, without the use of an "AnswerFile".

1. Copy the unzipped "Evaluate-STIG" Folder onto your Target Device, in the Directory of your choice.
2. Run PowerShell as Administrator
3. Parse to the "Evaluate-STIG" folder location in PowerShell: Type in "**cd [Directory Path]**" and then hit "**Enter**"
    a. Ex: "**cd C:\Users\Guest\Evaluate-STIG**"
4. **IMPORTANT:** This next change is temporary, and must be switched back to the way it was after the scan has completed.
5. Type in "**Get-ExecutionPolicy**", and press "**Enter**"
    a. **IMPORTANT:** Write down the output of this command, it will be adjusted accordingly later.
6. Type in "**Set-ExecutionPolicy RemoteSigned**", and press "**Enter**"
    a. Type in a capital "**A**", and then press "**Enter**"
7. For a Simplified Scan type in "**.\Evaluate-STIG.ps1 -ScanType [Unclassified or Classified] -OutputPath [The directory path for the output] -Output [CKL, CombinedCLK, STIGManager,... etc.] -SelectSTIG [Win10,MSEdge,... etc.]**"
    a. Example Input: .\Evaluate-STIG.ps1 -ComputerName "DESKTOP-4KAQQIP","PSDESKTOP-2" -ScanType Classified -OutputPath "C:\Admin\1. Checklists\2024\Q1" -Output CKL -SelectSTIG Win10,MSEdge"
    b. For the **-SelectSTIG** option, if you wanted to do more than one STIG: "**-SelectSTIG** Win10,MSEdge"
    c. **Note:** There is a lot of different command combinations, and supported STIGs for this tool. This is just one example of a combination of options for the command.
8. There should be a Red Bar at the top of the PowerShell Window, telling you the progress of the scan. Once it has completed, it will say:
    a. "**Applicable STIGs to process - #**"
    b. "**Done!**"
    c. "**Total Time : (time)**"
    d. "**Total CKLs in Results Directory : 2**"
    e. "**Results saved to C:\OutputPath provided beforehand**"
9. **IMPORTANT:** Type in "**Set-ExecutionPolicy Restricted**", and press "**Enter**"
    a. "**Restricted**" would be replaced by the output for "**Get-ExecutionPolicy**" that you wrote-down before.
    b. Type in a capital "**A**", and then press "**Enter**"
10. Open STIG Viewer as Administrator
11. Select the "STIG Explorer" Tab, then Click on "ChecklistOpen Checklist from File".
    a. Parse to your specified Evaluate-STIG "**OutputPath**" destination.
    b. Open the Folders "COMPUTERNAME", "Checklist", and then Select the Checklist (.CKL) File, and click "Open"
12. Now you're good to go! 🙂

ⓘ More details and specifics can be found in the "**/Evaluate-STIG_1.2307.2.zip/doc/Evaluate-STIG_UserGuide-1.2307.0.pdf**" file.

Useful Pages

- PowerShell (Command) Usage
  - **Pgs 8-12**
- AnswerFile Configuration
  - **Pgs 12-15**

What is Evaluate-STIG?

- Evaluate-STIG acts as a sort of SCAP Scan 2.0
- Evaluate-STIG is a PowerShell Script that works with its own library of supported STIG Checklists
  - In the Case of RHEL, both the Bash, "Evaluate-STIG_Bash.sh" and PowerShell, "Evaluate-STIG.ps1" files will work.
  - PowerShell 7 must be installed prior to utilizing both the Bash, and PowerShell Versions of Evaluate-STIG
    - The "Evaluate-STIG_Bash.sh" file is essentially a shell file for the "Evaluate-STIG.ps1" file. The "Evaluate-STIG_Bash.sh" file has PowerShell dependencies, which means that the Bash file will not be able to run, without PowerShell 7 installed.
- The user can choose to target one, multiple, or even all STIG Checklists at once
- Evaluate-STIG will look at everything on the target machine, from Group Policy Values, to Registry Values, and even more
- After it is done running with the provided STIG Checklists you set it to target, it will output a .CKL file, which can be opened in STIG Viewer
  - This .CKL file will also be the same as if you ran a SCAP Scan for the specific checklist, and also imported the .XCCDF data into the checklist
    - Yet, Evaluate-STIG will look at, and validate, way more than a traditional SCAP Scan
- You can also give it preset outputs for various different checklist items in an "AnswerFile"
  - This allows you have it auto-fill a response in the "Finding Details", for if something should have been found, if it's still open, or should not have been found
  - This can save a lot of time when it comes to filling out responses for checklist items that you know will fail, even though they have been addressed.

How to Get Evaluate-STIG

- **[REDACTED For Security Purposes]**

Red Hat Enterprise Linux - Evaluate-STIG - Table of Contents

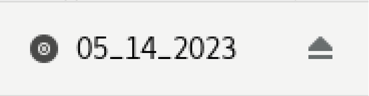
Red Hat Enterprise Linux Prerequisites

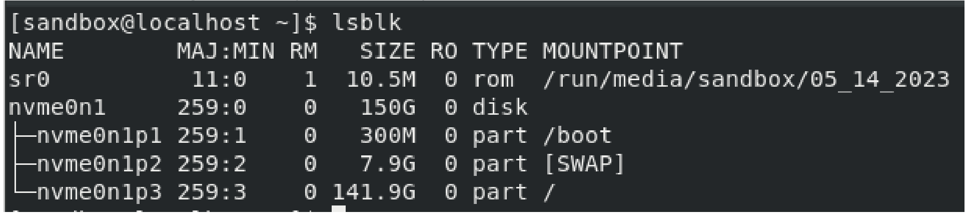
1. Make sure you have a Method of **Burning Files** to a .ISO File, on your **Primary Machine** (NOT YOUR TARGET MACHINE).
   a. The Application that I use for this is called "**AnyBurn**", which can be Downloaded here: http://www.anyburn.com/
2. Make sure your RHEL User Account is a member of the "**sudoers**" Group, or that you have Access to an Account which is a member of the "**sudoers**" Group.
3. Make sure you have **STIG Viewer 2.17** already Installed on the Target Device.
   a. If you do not have **STIG Viewer 2.17** Installed, and do not know how to Install it, there is a Guide for it here: How to Install STIG Viewer 2.17 on Red Hat Enterprise Linux (RHEL)
4. Make sure you have **PowerShell 7** already Installed on the Target Red Hat Linux Device.
   a. If you do not have **PowerShell 7** Installed, and do not know how to Install it, there is a Guide for it here: How to Install PowerShell 7 on Red Hat Enterprise Linux (RHEL)
5. Burn the "**Evaluate-STIG_1.2307.2.zip**" to a .ISO File.
   a. If you are using **AnyBurn**, you would choose the option "**Create image file from files/folders**".
   b. Click "**Add +**", Navigate to where you Downloaded the "**Evaluate-STIG_1.2307.2.zip**" File, Click on the File, and Click "**Add**".
   c. Click "**Next >**", Click the Folder Icon at the Top Right, Navigate to where you want to Save the .ISO File to, Type "**Evaluate-STIG_1.2307.2.zip.iso**" for the "**File name:**", and Click "**Save**".
   d. Click "**Create Now**", and when it has Finished Creating the .ISO File it will say, "**Creating image file finished successfully.**" in the "**Message**" Section.
   e. Now that the .ISO File has been Created, Navigate to where you Saved the .ISO File to.
      i. **If it is a Standalone Red Hat Enterprise Linux:** Copy the "**Evaluate-STIG_1.2307.2.zip.iso**" to an External Drive, and Insert the External Drive into the RHEL Machine.
      ii. **If it is a Red Hat Enterprise Linux VM:** Mount the "**Evaluate-STIG_1.2307.2.zip.iso**" in the VM Manager to your Virtual Machine.
         1. **For VMWare:** Click on your Red Hat Enterprise Linux Virtual Machine, and Click on "**Edit virtual machine settings**".
            a. Click on "**CD/DVD (SATA)**", Click on "**Use ISO image file:**", Click "**Browse...**", Navigate to where you Stored your "**Evaluate-STIG_1.2307.2.zip.iso**" File, and Click on it.
            b. Click "**Open**", Check the Box that says "**Connected**", and then Click "**OK**".
         2. **For VirtualBox:** Click on your Red Hat Enterprise Linux Virtual Machine, and Click on "**Settings**".
            a. Click on "**Storage**", Click on "**Controller: IDE**", Click on +, Click "**Optical Drive**", and Click on "**Add**".
            b. Navigate to where you Stored your "**Evaluate-STIG_1.2307.2.zip.iso**" File, and Click on it, and Click "**Open**".
            c. Click the Box that says "**Choose**" at the Bottom Right-Hand Corner, and then Click "**OK**".

6. Once you have met all of these Prerequisites, you can jump to the "**Installing Evaluate-STIG on Red Hat Enterprise Linux**" Section Below, to Utilize Evaluate-STIG!

Installing Evaluate-STIG on Red Hat Enterprise Linux

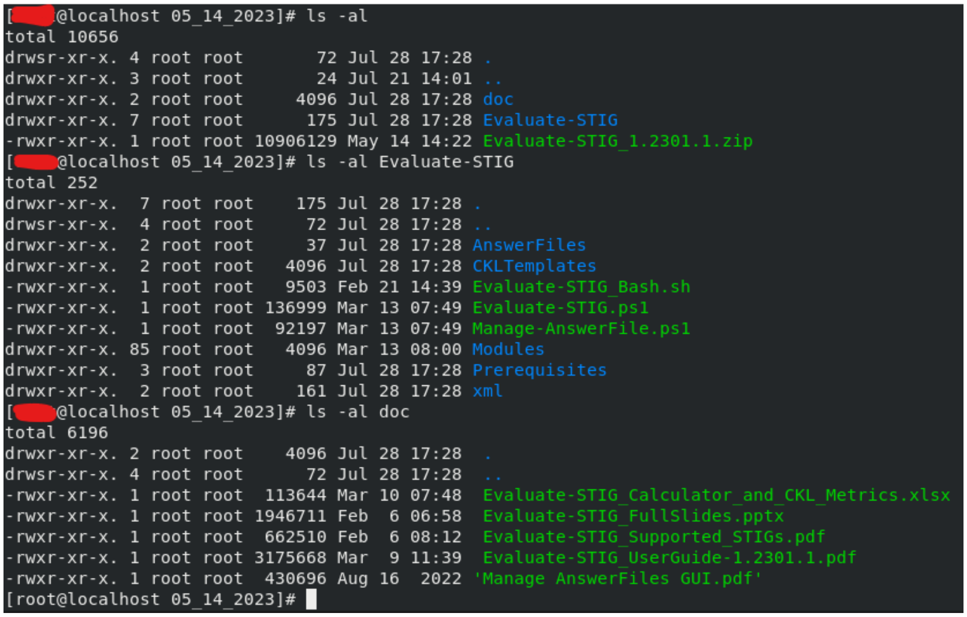
This guide assumes you have already imported the latest DISA STIG GPOs, and that you have STIG Viewer 2.17 already installed on the target device. This guide will focus on how to Install Evaluate-STIG on Red Hat Enterprise Linux.

1. Open the "**Files**" Application, and make sure that the "**Evaluate-STIG_1.2307.2.zip.iso**" File shows up in the File System.
    a. It will Appear as a CD, followed by the date you created the .ISO File, on the Left Side.
    b. 

2. Once you have verified that it is there, Open the "**Terminal**" Application, Type the Command "**lsblk**", and Press "**Enter**".
    a. Look for the "**MOUNTPOINT**" associated with the CD Drive that the File is attached to (The "**SIZE**" should be about "**10.5M**", if the "**Evaluate-STIG_1.2307.2.zip**" is the only File on there).
    b. 

```
[sandbox@localhost ~]$ lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sr0          11:0    1  10.5M  0 rom  /run/media/sandbox/05_14_2023
nvme0n1     259:0    0   150G  0 disk
├─nvme0n1p1 259:1    0   300M  0 part /boot
├─nvme0n1p2 259:2    0   7.9G  0 part [SWAP]
└─nvme0n1p3 259:3    0 141.9G  0 part /
```

    c. **Ex:** My "**Evaluate-STIG_1.2307.2.zip.iso**" File would be Located in "**/run/media/sandbox/05_14_2023**", since the File Size is "**10.5M**"
3. If you are logged into an account that does not have "**sudo**" Permissions: Open the "**Terminal**", Type in the Command, "**su [account name with sudo permissions]**", Press "**Enter**", Input the Password for that account, and then Press "**Enter**".
    a. WARNING: **BE CAREFUL OF WHAT YOU CHANGE/DO WHILST UTILIZING AN ACCOUNT WITH SUDO PERMISSIONS.**
    b. **Ex:** My Account with "**sudo**" permissions would be "**sandbox**", so I would Input the Command, "**su sandbox**"
4. Now that we are Currently Utilizing the an Administrator User Account, we can Create the Directory for Evaluate-STIG.
    a. Type in the Command "**mkdir /home/[username]/Desktop/Evaluate-STIG**", and Press "**Enter**"
5. Type in the Command "**cd /home/[username]/Desktop**", and Press "**Enter**"
    a. Type in the Command "**ls -al**", Press "**Enter**", and Verify that the "**/Evaluate-STIG**" Directory was Created Successfully.
6. Type in the Command "**sudo cp ["MOUNTPOINT" of "Evaluate-STIG_1.2307.2.zip.iso"] /home/[username]/Desktop/Evaluate-STIG**", and Press "**Enter**"
    a. **Ex:** My "**Evaluate-STIG_1.2307.2.zip.iso**" was mounted to "**/run/media/sandbox/05_14_2023**".
        i. So the Command I would Input is: "**sudo cp /run/media/sandbox/05_14_2023 /home/sandbox/Desktop/Evaluate-STIG**"
7. Type in the Command "**cd /home/[username]/Desktop/Evaluate-STIG**", and Press "**Enter**"
    a. **Ex:** "**cd /home/sandbox/Desktop/Evaluate-STIG**"
8. Type in the Command "**ls -al**", Press "**Enter**", and Verify that the "**[date that you created the .ISO]**" File was Copied Successfully.
9. Type in the Command "**sudo chmod 755 -R [the date you created the .ISO]**", Press "**Enter**"
    a. **Ex:** "**sudo chmod 755 -R 05_14_2023**"
    b. Type in the Command "**ls -al**", Press "**Enter**", and Verify that the "**Evaluate-STIG_1.2307.2.zip**" (or your Corresponding Evaluate-STIG Version) File was Copied Successfully.
10. Type in the Command "**sudo unzip Evaluate-STIG_1.2307.2.zip**" or "**sudo unzip [your Evaluate-STIG Version]**", and Press "**Enter**".
    a. This will Un-Zip your Evaluate-STIG .ZIP File, so that you can access the Files within.
11. Type in the Command "**sudo chmod 755 -R /home/[username]/Desktop/Evaluate-STIG/[the date you created the .ISO]**", Press "**Enter**"
    a. **Ex:** "**sudo chmod 755 -R /home/sandbox/Desktop/Evaluate-STIG/05_14_2023**"
12. Type in the Command "**ls -al**", Press "**Enter**", Type in the Command "**ls -al Evaluate-STIG**", Press "**Enter**", and Type in the Command "**ls -al doc** ", Press "**Enter**".
    a. Verify that the "**Evaluate-STIG**" and "**doc**" Folders have the Permissions "**drwxr-xr-x.**", as seen in the Screenshot Below.

b. 

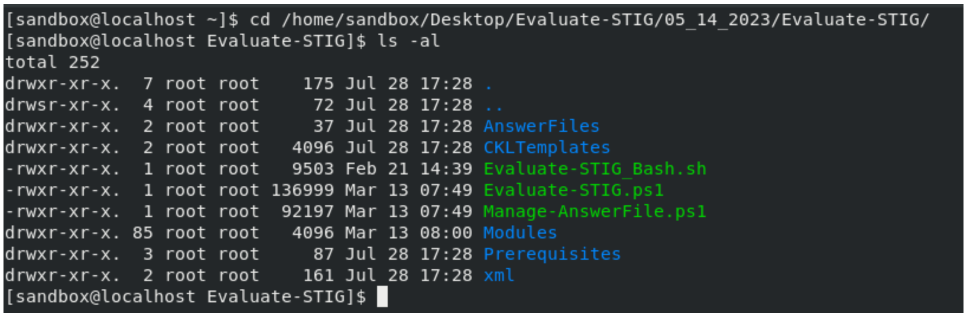
```
[[redacted]@localhost 05_14_2023]# ls -al
total 10656
drwsr-xr-x. 4 root root       72 Jul 28 17:28 .
drwxr-xr-x. 3 root root       24 Jul 21 14:01 ..
drwxr-xr-x. 2 root root     4096 Jul 28 17:28 doc
drwxr-xr-x. 7 root root      175 Jul 28 17:28 Evaluate-STIG
-rwxr-xr-x. 1 root root 10906129 May 14 14:22 Evaluate-STIG_1.2301.1.zip
[[redacted]@localhost 05_14_2023]# ls -al Evaluate-STIG
total 252
drwxr-xr-x.  7 root root    175 Jul 28 17:28 .
drwsr-xr-x.  4 root root     72 Jul 28 17:28 ..
drwxr-xr-x.  2 root root     37 Jul 28 17:28 AnswerFiles
drwxr-xr-x.  2 root root   4096 Jul 28 17:28 CKLTemplates
-rwxr-xr-x.  1 root root   9503 Feb 21 14:39 Evaluate-STIG_Bash.sh
-rwxr-xr-x.  1 root root 136999 Mar 13 07:49 Evaluate-STIG.ps1
-rwxr-xr-x.  1 root root  92197 Mar 13 07:49 Manage-AnswerFile.ps1
drwxr-xr-x. 85 root root   4096 Mar 13 08:00 Modules
drwxr-xr-x.  3 root root     87 Jul 28 17:28 Prerequisites
drwxr-xr-x.  2 root root    161 Jul 28 17:28 xml
[[redacted]@localhost 05_14_2023]# ls -al doc
total 6196
drwxr-xr-x. 2 root root    4096 Jul 28 17:28  .
drwsr-xr-x. 4 root root      72 Jul 28 17:28  ..
-rwxr-xr-x. 1 root root  113644 Mar 10 07:48  Evaluate-STIG_Calculator_and_CKL_Metrics.xlsx
-rwxr-xr-x. 1 root root 1946711 Feb  6 06:58  Evaluate-STIG_FullSlides.pptx
-rwxr-xr-x. 1 root root  662510 Feb  6 08:12  Evaluate-STIG_Supported_STIGs.pdf
-rwxr-xr-x. 1 root root 3175668 Mar  9 11:39  Evaluate-STIG_UserGuide-1.2301.1.pdf
-rwxr-xr-x. 1 root root  430696 Aug 16  2022 'Manage AnswerFiles GUI.pdf'
[root@localhost 05_14_2023]#
```

c. If this is the case, the Permissions for the "**~/Evaluate-STIG**" Directory have been properly configured.

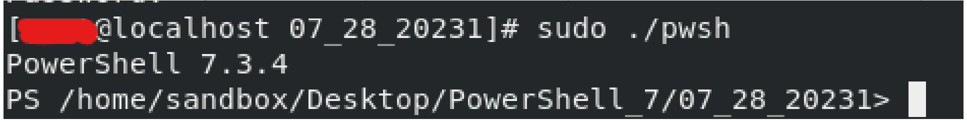
13. Type in the Command "**cd /home/[username]/Desktop/Evaluate-STIG**", and Press "**Enter**"
    a. **Ex:** "**cd /home/sandbox/Desktop/Evaluate-STIG**"
14. Next, make a Directory to Store Your .CKL Output Files: Type in the Command "**mkdir Checklists**", and Press "**Enter**".
15. Type in the Command "**ls -al**", and Verify that the "**Checklists**" Folder has been Created.
16. **IMPORTANT:** **If you switched into an account that has "sudo" Permissions:** **SWITCH BACK TO YOUR NORMAL USER ACCOUNT** - Type in the Command, "**su [your user account]**", and Press "**Enter**".
    a. **Ex:** My Normal/Personal User Account is called "**sandbox**", so I would Input the Command, "**su sandbox**"
17. Once you have met all of these Prerequisites, you can jump to the "**Simplified Step-by-Step Guide With PowerShell on RHEL (Without Use of an AnswerFile)**" Section Below, to Utilize Evaluate-STIG!

Simplified Step-by-Step Guide With PowerShell on RHEL (Without Use of an AnswerFile)

This guide assumes you have already imported the latest DISA STIG GPOs, that you have STIG Viewer 2.17 and PowerShell 7 already installed on the target device, that you have already completed the "**Installing Evaluate-STIG on Red Hat Enterprise Linux**" Portion of the Guide, and that you are currently signed into an account with "sudo" privileges. This guide will focus on how to use Evaluate-STIG, without the use of "AnswerFiles".

1. Type in the Command "**cd /home/[username]/Desktop/Evaluate-STIG/[the date you created the Evaluate-STIG .ISO]/Evaluate-STIG**", and Press "**Enter**"
    a. **Ex:** "**cd /home/sandbox/Desktop/Evaluate-STIG/05_14_2023/Evaluate-STIG**"
2. Type in the Command "**ls -al**", Press "**Enter**", and Verify that Folders, and their associated Permissions, Match those Seen in the Screenshot Below.
    a. 

```
[sandbox@localhost ~]$ cd /home/sandbox/Desktop/Evaluate-STIG/05_14_2023/Evaluate-STIG/
[sandbox@localhost Evaluate-STIG]$ ls -al
total 252
drwxr-xr-x.  7 root root    175 Jul 28 17:28 .
drwsr-xr-x.  4 root root     72 Jul 28 17:28 ..
drwxr-xr-x.  2 root root     37 Jul 28 17:28 AnswerFiles
drwxr-xr-x.  2 root root   4096 Jul 28 17:28 CKLTemplates
-rwxr-xr-x.  1 root root   9503 Feb 21 14:39 Evaluate-STIG_Bash.sh
-rwxr-xr-x.  1 root root 136999 Mar 13 07:49 Evaluate-STIG.ps1
-rwxr-xr-x.  1 root root  92197 Mar 13 07:49 Manage-AnswerFile.ps1
drwxr-xr-x. 85 root root   4096 Mar 13 08:00 Modules
drwxr-xr-x.  3 root root     87 Jul 28 17:28 Prerequisites
drwxr-xr-x.  2 root root    161 Jul 28 17:28 xml
[sandbox@localhost Evaluate-STIG]$
```

    b. If the Folder Contents, or Permissions, Don't Match those seen in the Screenshot Above, Repeat the "**Installing Evaluate-STIG on Red Hat Enterprise Linux**" Portion of the Guide.
3. Type in the Command "**cd /home/[username]/Desktop/PowerShell_7/[the date you created the .ISO]**", and Press "**Enter**"
    a. **Ex:** "**cd /home/sandbox/Desktop/PowerShell_7/07_28_20231**"
    b. If you Saved/Installed PowerShell 7 in a Different Directory, Utilize the Command, "**cd /[absolute path where you installed PowerShell 7]**", and Press "**Enter**"
4. Type in the Command, "**sudo ./pwsh**", and Enter your Password if Prompted, and PowerShell 7.3.4 should be Opened in the Terminal.
    a. 

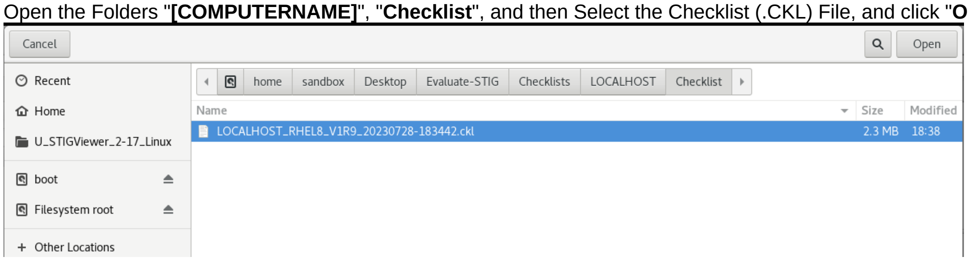
```
[[redacted]@localhost 07_28_20231]# sudo ./pwsh
PowerShell 7.3.4
PS /home/sandbox/Desktop/PowerShell_7/07_28_20231>
```

    b. The use of the "**sudo**" Command, is the Equivalent to running PowerShell 7 as Administrator, in Windows.
5. Parse to the "Evaluate-STIG" folder location in PowerShell: Type in "**cd /home/[username]/Desktop/Evaluate-STIG/[the date you created the Evaluate-STIG .ISO]/Evaluate-STIG**" and then hit "**Enter**"

a. Ex: "**cd /home/sandbox/Desktop/Evaluate-STIG/05_14_2023/Evaluate-STIG**"

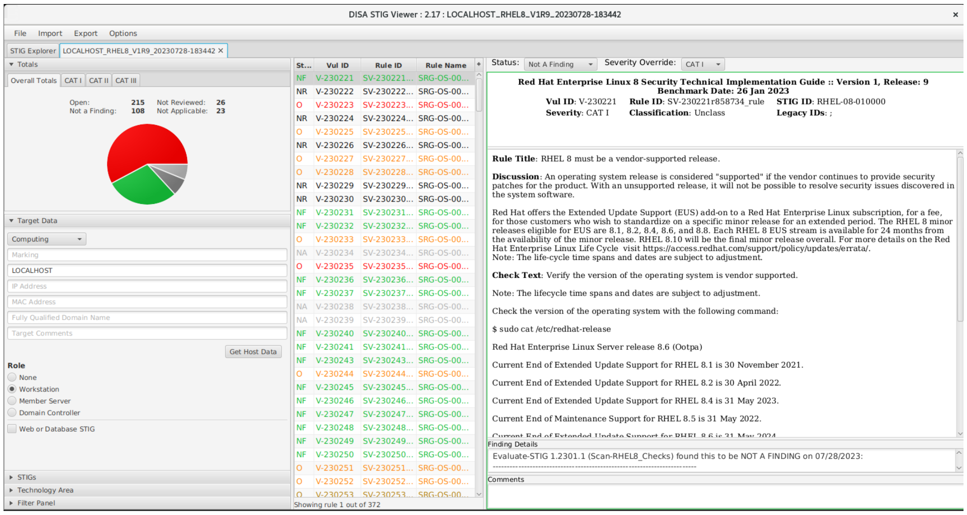
b. If you Saved/Installed PowerShell 7 in a Different Directory, Utilize the Command, "**cd /[absolute path where you installed PowerShell 7]**", and Press "**Enter**"

6. For a Simplified Scan type in "**./Evaluate-STIG.ps1 -ScanType [Unclassified or Classified] -OutputPath [The directory path for the output] -Output [CKL, CombinedCLK, STIGManager,... etc.] -SelectSTIG [RHEL8,FireFox... etc.]**"
   a. Example Input: "**./Evaluate-STIG.ps1 -ScanType Classified -OutputPath "/home/sandbox/Desktop/Evaluate-STIG" -Output CKL -SelectSTIG RHEL8**"
   b. For the **-SelectSTIG** option, if you wanted to do more than one STIG: "**-SelectSTIG RHEL8,FireFox**"
   c. **Note:** There is a lot of different command combinations, and supported STIGs for this tool. This is just one example of a combination of options for the command.
7. There should be a Red Bar at the top of the PowerShell Window, telling you the progress of the scan. Once it has completed, it will say:
   a. "**Applicable STIGs to process - #**"
   b. "**Done!**"
   c. "**Total Time : (time)**"
   d. "**Total CKLs in Results Directory : 1**"
   e. "**Results saved to /OutputPath provided beforehand**"
8. Exit out of the PowerShell Application: Type the Command "**exit**", and Press "**Enter**"
9. Open STIG Viewer as Administrator: Type in the Command "**cd /home/[username]/Desktop/STIG_Viewer/[the date you created the .ISO]/U_STIGViewer_2-17_Linux**", and Press "**Enter**"
   a. **Ex: "cd /home/sandbox/Desktop/STIG_Viewer/07_28_2023/U_STIGViewer_2-17_Linux"**
   b. If you Saved/Installed STIG Viewer in a Different Directory, Utilize the Command, "**cd /[absolute path where you installed STIG Viewer]**", and Press "**Enter**"
10. Now Type in the Command, "**sudo ./STIGViewer**", Press "**Enter**", Enter your Password if Prompted
    a. The use of the "**sudo**" Command, is the Equivalent to running STIG Viewer as Administrator, in Windows.
11. Once you are in the STIG Viewer Application, Click on "ChecklistOpen Checklist from File...".
    a. Parse to your specified Evaluate-STIG "**OutputPath**" destination.
    b. Open the Folders "**[COMPUTERNAME]**", "**Checklist**", and then Select the Checklist (.CKL) File, and click "**Open**"
    c.



LOCALHOST_RHEL8_V1R9_20230728-183442.ckl 2.3 MB 18:38

    d.



Open: 215 Not Reviewed: 26
Not a Finding: 108 Not Applicable: 23

Computing
Marking
LOCALHOST
IP Address
MAC Address
Fully Qualified Domain Name
Target Comments

Role
○ None
● Workstation
○ Member Server
○ Domain Controller

□ Web or Database STIG

St...	Vul ID	Rule ID	Rule Name
NF	V-230221	SV-230221...	SRG-OS-00...
NR	V-230222	SV-230222...	SRG-OS-00...
O	V-230223	SV-230223...	SRG-OS-00...
NR	V-230224	SV-230224...	SRG-OS-00...
O	V-230225	SV-230225...	SRG-OS-00...
NR	V-230226	SV-230226...	SRG-OS-00...
O	V-230227	SV-230227...	SRG-OS-00...
O	V-230228	SV-230228...	SRG-OS-00...
NR	V-230229	SV-230229...	SRG-OS-00...
NR	V-230230	SV-230230...	SRG-OS-00...
NF	V-230231	SV-230231...	SRG-OS-00...
NF	V-230232	SV-230232...	SRG-OS-00...
O	V-230233	SV-230233...	SRG-OS-00...
NA	V-230234	SV-230234...	SRG-OS-00...
O	V-230235	SV-230235...	SRG-OS-00...
NF	V-230236	SV-230236...	SRG-OS-00...
NF	V-230237	SV-230237...	SRG-OS-00...
NA	V-230238	SV-230238...	SRG-OS-00...
NA	V-230239	SV-230239...	SRG-OS-00...
NF	V-230240	SV-230240...	SRG-OS-00...
NF	V-230241	SV-230241...	SRG-OS-00...
NF	V-230243	SV-230243...	SRG-OS-00...
O	V-230244	SV-230244...	SRG-OS-00...
NF	V-230245	SV-230245...	SRG-OS-00...
NF	V-230246	SV-230246...	SRG-OS-00...
NF	V-230247	SV-230247...	SRG-OS-00...
NF	V-230248	SV-230248...	SRG-OS-00...
NF	V-230249	SV-230249...	SRG-OS-00...
NF	V-230250	SV-230250...	SRG-OS-00...
O	V-230251	SV-230251...	SRG-OS-00...
O	V-230252	SV-230252...	SRG-OS-00...
O	V-230253	SV-230253...	SRG-OS-00...

Showing rule 1 out of 372

Status: Not A Finding Severity Override: CAT I

Red Hat Enterprise Linux 8 Security Technical Implementation Guide :: Version 1, Release: 9
Benchmark Date: 26 Jan 2023

Vul ID: V-230221 **Rule ID**: SV-230221r858734_rule **STIG ID**: RHEL-08-010000
Severity: CAT I **Classification**: Unclass **Legacy IDs**: ;

Rule Title: RHEL 8 must be a vendor-supported release.

Discussion: An operating system release is considered "supported" if the vendor continues to provide security patches for the product. With an unsupported release, it will not be possible to resolve security issues discovered in the system software.

Red Hat offers the Extended Update Support (EUS) add-on to a Red Hat Enterprise Linux subscription, for a fee, for those customers who wish to standardize on a specific minor release for an extended period. The RHEL 8 minor releases eligible for EUS are 8.1, 8.2, 8.4, 8.6, and 8.8. Each RHEL 8 EUS stream is available for 24 months from the availability of the minor release. RHEL 8.10 will be the final minor release overall. For more details on the Red Hat Enterprise Linux Life Cycle visit https://access.redhat.com/support/policy/updates/errata/.
Note: The life-cycle time spans and dates are subject to adjustment.

Check Text: Verify the version of the operating system is vendor supported.

Note: The lifecycle time spans and dates are subject to adjustment.

Check the version of the operating system with the following command:

```
$ sudo cat /etc/redhat-release

Red Hat Enterprise Linux Server release 8.6 (Ootpa)
```

Current End of Extended Update Support for RHEL 8.1 is 30 November 2021.

Current End of Extended Update Support for RHEL 8.2 is 30 April 2022.

Current End of Extended Update Support for RHEL 8.4 is 31 May 2023.

Current End of Maintenance Support for RHEL 8.5 is 31 May 2022.

Current End of Extended Update Support for RHEL 8.6 is 31 May 2024.

Finding Details

Evaluate-STIG 1.2301.1 (Scan-RHEL8_Checks) found this to be NOT A FINDING on 07/28/2023:

Comments

12. The Evaluate-STIG Checklist File Should have opened, and you should be good to go! 🙂

More details and specifics can be found in the "**/Evaluate-STIG_1.2307.2.zip/doc/Evaluate-STIG_UserGuide-1.2307.0.pdf**" file.

Useful Pages

- PowerShell (Command) Usage
  - **Pgs 8-12**
- AnswerFile Configuration
  - **Pgs 12-15**

What are Shortname Abbreviations in Evaluate-STIG?

- When specifying STIG Checklists in Evaluate-STIG, you can use the Full-Name of the STIG Checklist, or the "Shortname" abbreviation for it.
- These Shortname abbreviations can be utilized in Evaluate-STIG Scans, and to configure Evaluate-STIG AnswerFiles.
- You can access the full list of Names, Shortnames, Versions, and associated STIG Checklist Templates for each STIG Checklist item, by running Evaluate-STIG, with the "-ListSupportedProducts" option.
    - Example: **.\Evaluate-STIG.ps1** **-ListSupportedProducts**

Evaluate-STIG Supported STIG Checklist Names, Shortnames, Versions, and STIG Checklist Templates

STIG Checklist Name	Evaluate-STIG Shortname	STIG Checklist Version	STIG Checklist Template
Active Directory Domain	ADDomain	V3R3	ADDomain.ckl
Active Directory Forest	ADForest	V2R8	ADForest.ckl
Adobe Acrobat Pro XI	AdobeAcrobatProXI	V1R2	AdobeAcrobatProXI.ckl
Adobe Acrobat Professional DC Classic	AdobeAcrobatProDCClassic	V2R1	AdobeAcrobatProDCClassic.ckl
Adobe Acrobat Professional DC Continuous	AdobeAcrobatProDCContinuous	V2R1	AdobeAcrobatProDCContinuous.ckl
Adobe Reader DC Classic	AdobeReaderDCClassic	V2R1	AdobeReaderDCClassic.ckl
Adobe Reader DC Continuous	AdobeReaderDCContinuous	V2R1	AdobeReaderDCContinuous.ckl
Apache 2.4 Server Unix	Apache24SvrUnix	V2R5	Apache24SvrUnix.ckl
Apache 2.4 Server Windows	Apache24SvrWin	V2R3	Apache24SvrWin.ckl
Apache 2.4 Site Unix	Apache24SiteUnix	V2R4	Apache24SiteUnix.ckl
Apache 2.4 Site Windows	Apache24SiteWin	V2R1	Apache24SiteWin.ckl
Apache Tomcat Application Server	ApacheTomcatAS	V2R5	ApacheTomcatAS.ckl
CentOS 7	CentOS7	V3R12	CentOS7.ckl
Cisco IOS XE Router NDM	CiscoXERtrNDM	V2R7	CiscoXERouterNDM.ckl
Cisco IOS XE Switch L2S	CiscoXESwtchL2S	V2R4	CiscoXESwitchL2S.ckl
Cisco IOS XE Switch NDM	CiscoXESwtchNDM	V2R6	CiscoXESwitchNDM.ckl
Google Chrome	Chrome	V2R8	Chrome.ckl
IIS 10.0 Server	IIS10Server	V2R9	IIS10Server.ckl
IIS 10.0 Site	IIS10Site	V2R8	IIS10Site.ckl
IIS 8.5 Server	IIS85Server	V2R6	IIS85Server.ckl
IIS 8.5 Site	IIS85Site	V2R8	IIS85Site.ckl
Internet Explorer 11	IE11	V2R4	IE11.ckl
McAfee ENS 10x Local	McAfeeENS10xLocal	Template missing *	(CUI)_McAfeeENS10xLocal.ckl
McAfee VirusScan 8.8 Local Client	McAfeeVS88	V6R1	McAfeeVS88.ckl
Microsoft .NET Framework 4	DotNET4	V2R2	DotNET4.ckl
Microsoft Access 2013	MSAccess2013	V1R6	MSAccess2013.ckl
Microsoft Access 2016	MSAccess2016	V1R1	MSAccess2016.ckl
Microsoft Defender Antivirus	MSDefender	V2R4	MsDefender.ckl
Microsoft Edge	MSEdge	V1R7	MSEdge.ckl
Microsoft Excel 2013	MSExcel2013	V1R7	MSExcel2013.ckl
Microsoft Excel 2016	MSExcel2016	V1R2	MSExcel2016.ckl
Microsoft Exchange 2016 Edge Transport Server	MSExchange2016EdgeTP	V2R4	MSExchange2016EdgeTP.ckl
Microsoft Exchange 2016 Mailbox Server	MSExchange2016MB	V2R4	MSExchange2016MB.ckl
Microsoft Groove 2013	MSGroove2013	V1R3	MSGroove2013.ckl
Microsoft InfoPath 2013	MSInfoPath2013	V1R5	MSInfoPath2013.ckl
Microsoft Lync 2013	MSLync2013	V1R4	MSLync2013.ckl
Microsoft Office 365	MSOffice365	V2R10	MSOffice365.ckl
Microsoft Office System 2013	MSOfficeSystem2013	V2R1	MSOfficeSystem2013.ckl
Microsoft Office System 2016	MSOfficeSystem2016	V2R2	MSOfficeSystem2016.ckl

Microsoft OneDrive	MSOneDrive	V2R3	MSOneDrive.ckl
Microsoft OneNote 2013	MSOneNote2013	V1R3	MSOneNote2013.ckl
Microsoft OneNote 2016	MSOneNote2016	V1R2	MSOneNote2016.ckl
Microsoft Outlook 2013	MSOutlook2013	V1R13	MSOutlook2013.ckl
Microsoft Outlook 2016	MSOutlook2016	V2R3	MSOutlook2016.ckl
Microsoft PowerPoint 2013	MSPowerPoint2013	V1R6	MSPowerPoint2013.ckl
Microsoft PowerPoint 2016	MSPowerPoint2016	V1R1	MSPowerPoint2016.ckl
Microsoft Project 2013	MSProject2013	V1R4	MSProject2013.ckl
Microsoft Project 2016	MSProject2016	V1R1	MSProject2016.ckl
Microsoft Publisher 2013	MSPublisher2013	V1R5	MSPublisher2013.ckl
Microsoft Publisher 2016	MSPublisher2016	V1R3	MSPublisher2016.ckl
Microsoft SharePoint Designer 2013	MSSPDesigner2013	V1R3	MSSPDesigner2013.ckl
Microsoft Skype for Business 2016	MSSkype2016	V1R1	MSSkype2016.ckl
Microsoft SQL Server 2014 Database	SQL2014DB	V1R6	SQL2014DB.ckl
Microsoft SQL Server 2014 Instance	SQL2014Instance	V2R3	SQL2014Instance.ckl
Microsoft SQL Server 2016 Database	SQL2016DB	V2R6	SQL2016DB.ckl
Microsoft SQL Server 2016 Instance	SQL2016Instance	V2R10	SQL2016Instance.ckl
Microsoft Visio 2013	MSVisio2013	V1R4	MSVisio2013.ckl
Microsoft Visio 2016	MSVisio2016	V1R1	MSVisio2016.ckl
Microsoft Word 2013	MSWord2013	V1R6	MSWord2013.ckl
Microsoft Word 2016	MSWord2016	V1R1	MSWord2016.ckl
Mozilla Firefox	Firefox	V6R5	Firefox.ckl
Oracle Java JRE 8 for Unix	JavaJRE8Unix	V1R3	JavaJRE8_Unix.ckl
Oracle Java JRE 8 for Windows	JavaJRE8Windows	V2R1	JavaJRE8_Windows.ckl
Oracle Linux 7	Oracle7	V2R12	Oracle7.ckl
Oracle Linux 8	Oracle8	V1R7	Oracle8.ckl
PostgreSQL 9.x	PgSQL9x	V2R3	PgSQL9x.ckl
Red Hat Enterprise Linux 7	RHEL7	V3R12	RHEL7.ckl
Red Hat Enterprise Linux 8	RHEL8	V1R11	RHEL8.ckl
Ubuntu 16.04	Ubuntu16	V2R3	Ubuntu16.ckl
Ubuntu 18.04	Ubuntu18	V2R11	Ubuntu18.ckl
Ubuntu 20.04	Ubuntu20	V1R9	Ubuntu20.ckl
VMware Horizon 7.13 Agent	HorizonAgent	V1R1	VMwareHorizon7-13Agent.ckl
VMware Horizon 7.13 Client	HorizonClient	V1R1	VMwareHorizon7-13Client.ckl
VMware Horizon 7.13 Connection Server	HorizonConnectionServer	V1R1	VMwareHorizon7-13ConnectionServer.ckl
Windows 10	Win10	V2R7	Win10.ckl
Windows 11	Win11	V1R4	Win11.ckl
Windows Firewall	WinFirewall	V2R1	WinFirewall.ckl
Windows Server 2008 R2 MS	WinServer2008R2MS	V1R33	WinServer2008R2MS.ckl
Windows Server 2012 DC	WinServer2012DC	V3R6	WinServer2012DC.ckl
Windows Server 2012 MS	WinServer2012MS	V3R6	WinServer2012MS.ckl
Windows Server 2016	WinServer2016	V2R6	WinServer2016.ckl
Windows Server 2019	WinServer2019	V2R7	WinServer2019.ckl
Windows Server 2022	WinServer2022	V1R3	WinServer2022.ckl

* Separate download not included with Evaluate-STIG distributable.

More details and specifics can be found in the "**/Evaluate-STIG_1.2307.2.zip/doc/Evaluate-STIG_UserGuide-1.2307.0.pdf**" file.

Useful Pages

- PowerShell (Command) Usage
  - **Pgs 8-12**
- AnswerFile Configuration
  - **Pgs 12-15**

The greatest source of information for Evaluate-STIG can be found in the "Evaluate-STIG_UserGuide". A copy of this can be found at the bottom of this page. The purpose of this page is to address basic, and specific, use-cases for Evaluate-STIG, which the User Guide may not cover in-depth.

User Guide

Evaluate-STIG..._1.2401.3.pdf

Related articles

- Evaluate-STIG Guide & Documentation