

Extending the Reach of Kerberos Authentication from SAS® Viya® 3.4 to Apache Hadoop

Stuart J Rogers, SAS Institute Inc.

ABSTRACT

Strong authentication using techniques such as Kerberos is becoming an IT security requirement for many organizations. SAS® Viya® 3.4 supports the option to delegate Kerberos credentials throughout the environment and onto your Apache Hadoop distribution. Doing so enables you to provide strong authentication both into and out of your SAS Viya 3.4 environment. This paper describes the steps that a SAS administrator and IT security specialist need to complete in order to enable strong authentication both into and out of a SAS Viya 3.4 environment.

INTRODUCTION

The use of Kerberos authentication across organizations is becoming more common. This paper will show you how your SAS Viya 3.4 deployment can use Kerberos authentication, giving you a better understanding of how Kerberos is used throughout your environment. We illustrate how Kerberos is used to authenticate into your environment, between SAS Viya processes, and out from the SAS Viya environment to your secured Hadoop or other third-party data sources.

We will focus heavily on the prerequisites that must be completed correctly. Most issues with Kerberos configuration are traced back to issues with completing the prerequisite steps. Correctly completing the prerequisites will be the most important step in getting Kerberos authentication working for your environment.

Then we will examine in detail the steps that you should complete to configure Kerberos authentication for your SAS Viya 3.4 environment. We will look at the configuration for both Linux and Microsoft Windows based environments. Highlighting the similarities and differences in the configuration process.

Next, we shall discuss how scheduling can require additional considerations to the planning for your end-to-end Kerberos implementation. We will show the different ways, you can make scheduling as successful as any other part of your environment. Finally, we discuss the other two clients for your environment, with the SAS Administration CLI and SAS Visual Analytic App. We show how these fully integrate with your Kerberos configuration.

KERBEROS WITH SAS VIYA 3.4 OVERVIEW

Before examining the prerequisites and configuration in detail, this paper will explain how Kerberos authentication is used in the SAS Viya 3.4 environment. Kerberos is the only supported browser authentication mechanism with SAS Viya 3.4 deployments on Microsoft Windows. Linux SAS Viya 3.4 deployments support additional authentication mechanisms with SAS Logon Manager.

KERBEROS AND SAS LOGON MANAGER

Kerberos authentication with SAS Logon Manager provides end users with Single Sign-On from their desktop, where the browser is running. This is sometimes referred to as Integrated Windows Authentication (IWA). This type of authentication enables the end user to access the SAS Viya 3.4 visual interfaces without being prompted for any credentials. However, it is important to remember that the Identities microservice will still be connecting

to the LDAP provider, to look up identity (such as email or name) and group information. The Kerberos authentication option completely replaces the option to use the default LDAP provider for the SAS Logon Manager. Introduced with SAS Viya 3.3 is the option to delegate the credentials to SAS Logon Manager to make them available to additional processes.

Configuring Kerberos authentication for SAS Logon Manager replaces all other browser-based authentication mechanisms. Once Kerberos is configured, it is the only mechanism that can be used in the browser to access the SAS Viya environment. Other access routes are not impacted. Therefore, the SAS Administration Command-Line Interface (CLI) will still correctly operate with a user name and password that have been validated by the LDAP provider.

When configured for Kerberos, SAS Logon Manager changes from validating the user name and password with LDAP to using Kerberos. So, a delegated Kerberos credential can be generated through the other access routes. This enables Kerberos delegation for the SAS Administration CLI tools and the SAS mobile application.

The delegated Kerberos credential is stored in the Credentials microservice, whether it comes from the web browser or the Kerberos authenticated user name and password. Anything stored with the Credentials microservice is encrypted using AES encryption and stored in the SAS Infrastructure Data Server. Authorization rules enable only the original end-user access to the stored credentials. The delegated credentials are never transported back to the client browser and are consumed only by other services within the SAS Viya environment. On Linux, all communications among the services are encrypted using HTTPS. On Microsoft Windows, all the services run on a single host, so the information is never sent outside that host.

KERBEROS AND SAS CLOUD ANALYTIC SERVICES (CAS)

Kerberos authentication can be leveraged by the CAS server in two ways: to authenticate to CAS, or to authenticate from a CAS session to a third-party data source, such as Hadoop.

By default, on Linux environments CAS sessions always run as the operating system account that is used to launch the CAS controller—typically the cas account. However, creating a custom group with an ID of CASHostAccountRequired means that members of the custom group will have their CAS sessions run as individual user accounts, typically as themselves. In Windows environments the CAS sessions always run as individual accounts, and the CASHostAccountRequired custom group is not required.

In Linux environments, the CAS sessions running as the cas account can access third-party data sources using Kerberos authentication. This is enabled by providing the CAS controller with a Kerberos keytab. When the CAS session is launched, the Kerberos keytab is used to initialize a Kerberos credential for the principal in the keytab. All sessions will then access the third-party data source using the single principal from the Kerberos Keytab. This default use case is illustrated in Figure 1.

The use case shown in Figure 1 will also be used by any services that directly access the third-party data source. For example, SAS® Model Studio creates internal objects in the third-party data source for storing model-processing information.

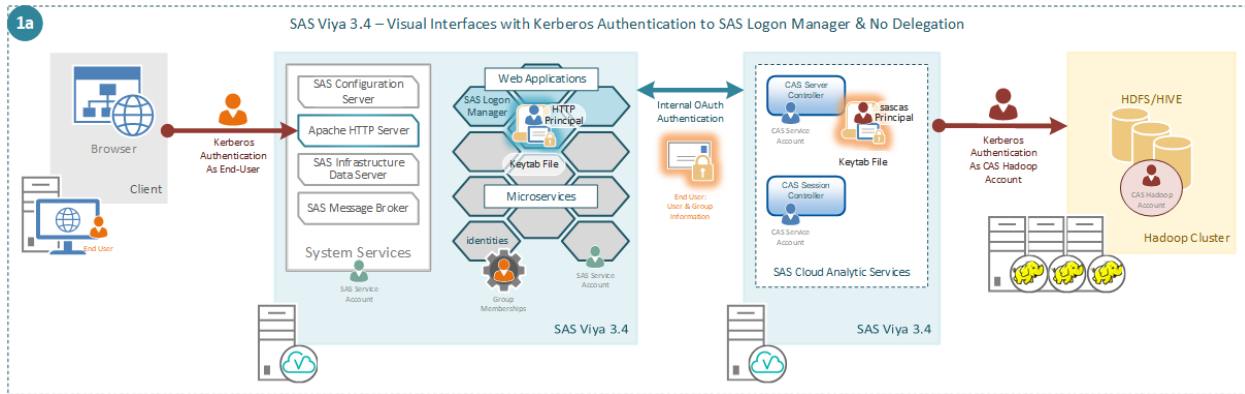


Figure 1. CAS Default Session Launch

With a Linux environment, to provide individual Kerberos access to the third-party data source, the end users must be members of the CASHostAccountRequired custom group. Your choice then is how to authenticate the end users to the CAS controller. If your users are using the SAS Viya 3.4 visual interfaces, they have two options. They can use the delegated Kerberos credentials stored by SAS Logon Manager, which is the preferred mechanism illustrated in Figure 2. Here, the delegated Kerberos credentials are used to authenticate to the CAS controller and are delegated onto the CAS session.

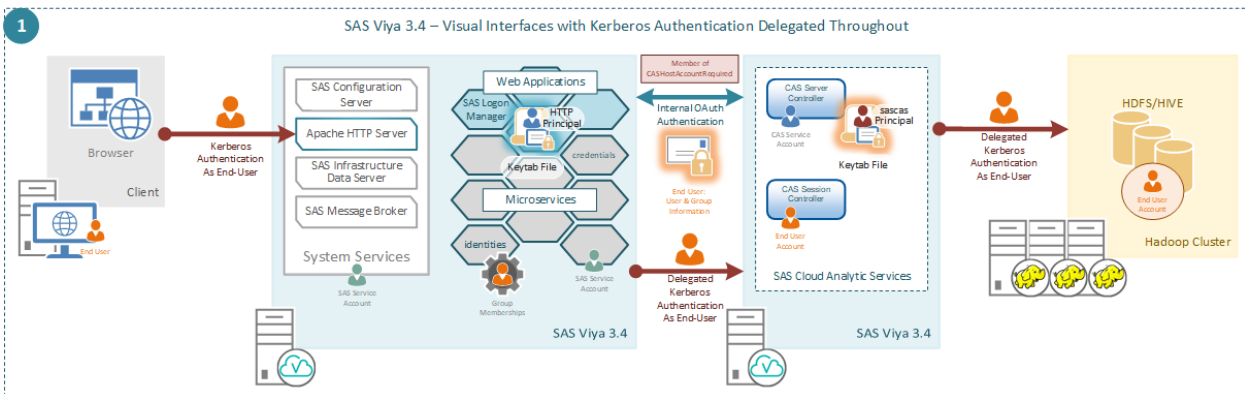


Figure 2. CAS with Kerberos Delegation

On Linux, end users could store a user name and password via SAS Environment Manager in the DefaultAuth authentication domain. The user name and password will be used to authenticate to the CAS controller and are validated through the Pluggable Authentication Module (PAM) stack on the host. If the PAM stack is integrated with Kerberos, a Kerberos Ticket-Granting Ticket (TGT) is stored in a credential cache on the file system and picked up by the CAS session. This option is illustrated in Figure 3.

The use case shown in Figure 3 assumes that the stored user name and password in the DefaultAuth authentication domain are individual to the end user. However, this could be a shared account associated with a group. A single end user must only have access to either an individual stored credential or to a group credential, but not to both.

In Microsoft Windows environments, Kerberos is the only supported authentication mechanism. Therefore, in Windows environments, Kerberos delegation through SAS Logon Manager must be used. In Windows environments the CAS session always runs as the end user and Kerberos is used to authenticate to the CAS controller, as shown in Figure 2. If the Kerberos credentials stored by SAS Logon Manager have expired, in a Windows environment CAS is also able to leverage the stored credentials, as shown in Figure 3.

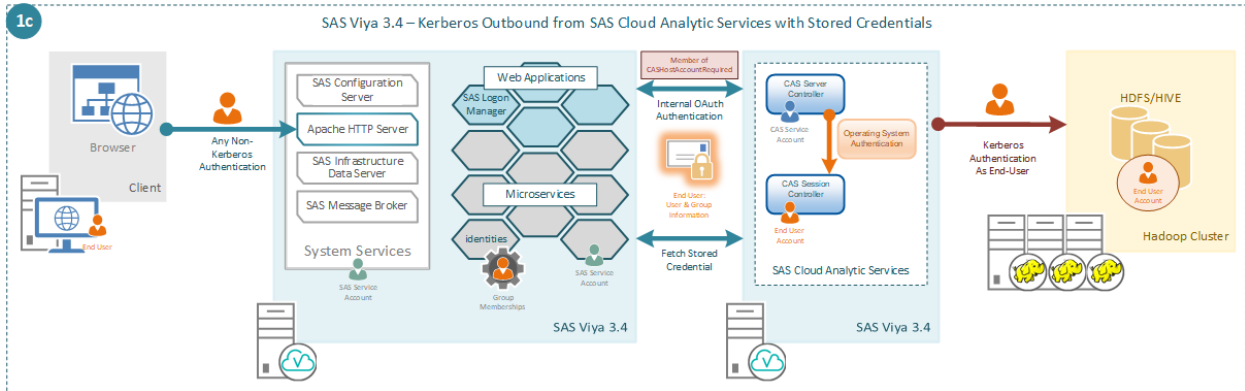


Figure 3. CAS with Stored Credentials

KERBEROS, SAS LAUNCHER, AND COMPUTE SERVER

The SAS Compute Server can leverage Kerberos authentication in two ways: either to authenticate to the SAS Launcher Server to launch a SAS Compute Server session, or to authenticate from a SAS Compute Server session to a third-party data source, such as Hadoop. The SAS Launcher Server and SAS Compute Server are always accessed via a SAS Viya 3.4 visual interface, such as SAS® Studio 5.1. The SAS Compute Server session always runs as an individual user account.

On Linux, the default configuration of the SAS Launcher Server uses a direct launch of the session as the end user logged in to the SAS Viya visual interfaces. This means that in the default scenario, no Kerberos credentials are available to the SAS Compute Server session. To provide Kerberos credentials to the SAS Compute Server session, you have two options. You can configure Kerberos delegation, which is the recommended approach, illustrated in Figure 4. With Kerberos delegation, the Kerberos credentials that are stored by SAS Logon Manager are used to authenticate the SAS Compute Server session through the SAS Launcher Server and are delegated onto that session.

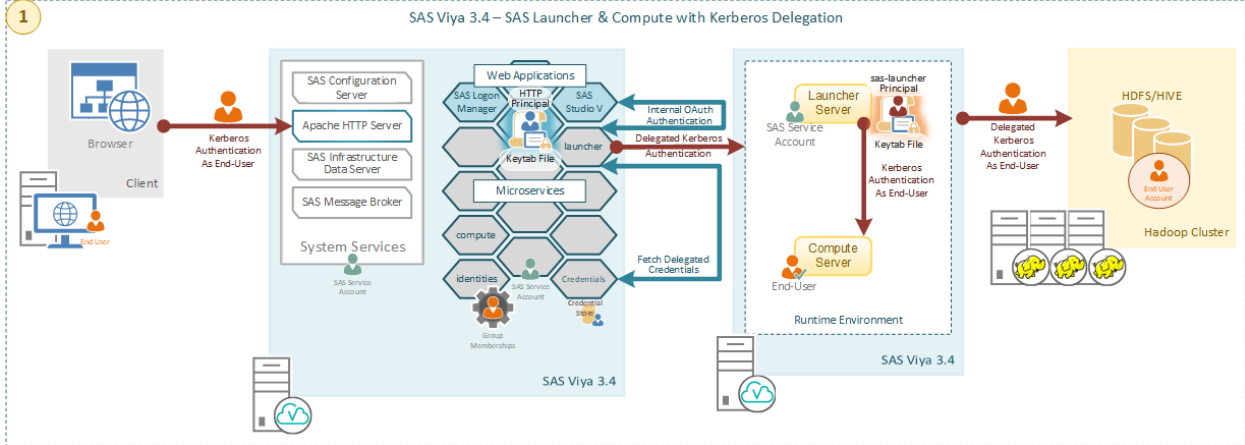


Figure 4. SAS Compute Server with Kerberos Delegation

As an alternative, on Linux, a user name and password can be stored against the DefaultAuth authentication domain in SAS Environment Manager. The stored user name and password can then be used to authenticate the SAS Compute Server session through the SAS Launcher Server. The user name and password are validated through the PAM stack, and if PAM is integrated with Kerberos, a Kerberos TGT will be available to the SAS Compute Server session. This use case is shown in Figure 5, which illustrates the user name and

password being stored for an individual. As is the case with CAS, this could be a shared user name and password that are associated with a group.

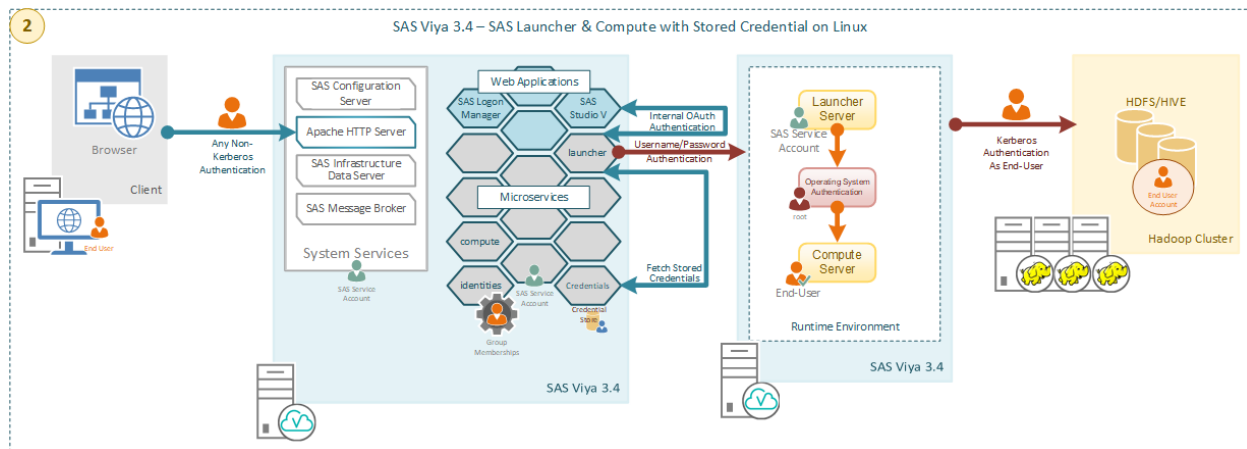


Figure 5. SAS Compute Server with Stored Credential

In Microsoft Windows environments, Kerberos delegation is the only supported authentication mechanism. So Microsoft Windows environments must always follow the use case illustrated in Figure 4.

PREREQUISITES

Now that we have reviewed how Kerberos authentication can be used in your SAS Viya 3.4 environment, we can move on to discuss the prerequisites. Issues with the prerequisites are the most common causes of implementation failures.

SAS Viya is supported on both Linux and Microsoft Windows operating systems. The prerequisites for Kerberos authentication are slightly different, depending on the operating system. Therefore, we will separately discuss the prerequisites for each operating system.

Also, SAS Viya 3.4 on Linux supports either Microsoft Active Directory or MIT Kerberos for the Kerberos Key Distribution Center (KDC). Although most prerequisites are the same for both KDC types, we will call out any specific differences related to either Microsoft Active Directory or MIT Kerberos.

You need to have Kerberos authentication for three separate services: SAS Logon Manager, CAS, and SAS Launcher Server. Therefore, SAS recommends that you treat each of these services separately. You must complete the prerequisite steps and configuration for all three services to have a correctly operating SAS Viya 3.4 environment.

LINUX PREREQUISITES

If you use Microsoft Active Directory as your KDC, you will need service accounts for the three services (SAS Logon Manager, CAS, and SAS Launcher Server). Associated with the service accounts will be their User Principal Names, Service Principal Names, and Kerberos Keytab files.

Or, if you use MIT Kerberos as the KDC, you will be concerned with only the Service Principal Names and Kerberos Keytab files.

Service Accounts

The service accounts for the three services do not require any elevated security privileges. These service accounts can be regular user accounts in Microsoft Active Directory.

However, to simplify ongoing management, you might consider extending the password cycle for these accounts. A minor outage is required each time the password of these service accounts is reset.

The names of these services accounts can be anything that fits your organization's standards. These service accounts will not be running any operating system processes in SAS Viya. However, for CAS, in some cases the service account will be the account accessing the third-party data source, as shown in Figure 1.

User Principal Names

In Microsoft Active Directory's implementation of the Kerberos authentication protocol, users are uniquely identified by their User Principal Name. Kerberos credentials can be generated only for a User Principal Name. The User Principal Name for the service accounts for SAS Logon Manager and SAS Launcher Server can be any value that fits your organization's standards.

The User Principal Name for CAS is important. The CAS server will initialize Kerberos credentials for a number of sessions using this User Principal Name, as shown in Figure 1. SAS recommends making the User Principal Name for CAS the same as the Service Principal Name.

Service Principal Names

A Service Principal Name (SPN) uniquely identifies an instance of a service running on a specific host. An SPN has the format <Service Class>/<Fully Qualified Hostname>@<Kerberos Realm>, where the three service classes are:

- SAS Logon Manager = HTTP
- CAS = sascas
- SAS Launcher Server = sas-launcher

The fully qualified host name corresponds to the host where the client accesses the service. For SAS Logon Manager, this value will not correspond to the hosts where instances of the SAS Logon Manager process are running. Instead, this host will be the HTTP Reverse Proxy to which your end users connect. By default, this will be an instance of the Apache HTTP Server, but this host could be a load-balancing proxy in front of the Apache HTTP Server.

For CAS, this will be the hosts where the CAS controller is running. In Linux environments, SAS supports a backup controller, so a single instance of CAS could have two different hosts running a controller process.

SAS supports horizontal scaling of the SAS Launcher Server and SAS Compute Server on Linux. As a result, there could be multiple hosts for the SPNs for the SAS Launcher Server.

In summary, on Linux, you might have the following setup:

- SAS Logon Manager, one HTTP/*Apache-HTTP-Server-host-name* SPN
- CAS, two *sascas/CAS-Controller-host-names* SPNs
- SAS Launcher Server, multiple *sas-launcher/SAS-Compute-Server-host-names* SPNs

If you are using Microsoft Active Directory as your KDC, the SPNs must be manually registered against the respective service accounts. The Microsoft Windows `setspn` command-line tool can be used to perform this step. Here is an example:

```
setspn -s HTTP/mywebserver.company.com sas-logon
```

The `setspn` command requires Write access to the service account in Microsoft Active Directory, which might mean that you need a Domain Administrator to run the commands for you.

If you are using MIT Kerberos as your KDC, you will just be registering the principal names. MIT Kerberos provides the `kadmin` tool, which you can use to register the principal names:

```
kadmin -q "addprinc -randkey +ok_as_delegate HTTP/mywebserver.company.com"
```

More information about the `kadmin` tool is available in the MIT Kerberos documentation (MIT 2018).

Kerberos Keytab Files

The Kerberos keytab file, as explained in the Kerberos documentation (MIT 2018), stores long-term keys for one or more principals. For your SAS Viya environment on Linux, you will need a minimum of three Kerberos keytab files, one for each service account. If the services are running on more than one host, you will need to copy the Kerberos keytab file to each host where the service is running.

When using Microsoft Active Directory for your KDC, SAS recommends that the Kerberos keytab file contain both the User Principal Names and Service Principal Names associated with the service account. For SAS Logon Manager, this can be very straightforward; if the User Principal Name is the same as the Service Principal Name, only a single principal is required in the Kerberos keytab file. For CAS and SAS Launcher Server, if you are running them on multiple hosts, multiple principals are required in the keytab file.

The Kerberos long-term key is generated from the password of the associated Microsoft Active Directory account and a salt. The salt is normally based on the User Principal Name of the account. Therefore, even if the password remains the same but the User Principal Name is changed, the long-term key will be different. The long-term key uses an encryption type. It is important for the Kerberos keytab to contain a matching encryption type to the service ticket presented to the service.

Using MIT Kerberos for your KDC simplifies matters because there is no distinction between a User Principal Name and a Service Principal Name; there are just principal names. Therefore, the Kerberos keytab will contain only what we have been referring to as Service Principal Names or SPNs.

As discussed in a paper by Mike Roda (Roda 2016), there are several ways to create a Kerberos keytab file. SAS recommends using the Linux utility `ktutil` because it does not change any attributes in the KDC. The `ktutil` utility also provides full control over the encryption types, which are included in the Kerberos keytab. Output 1 provides an example of creating the Kerberos keytab for SAS Logon Manager.

Using a key version number of zero will ensure that any checking of the key version number will be skipped. Notice that you will need to specify the password of the service account or principal to create the Kerberos keytab with `ktutil`. Additional details about the `ktutil` commands can be found in the MIT Kerberos documentation (MIT 2018).

```
$ ktutil
ktutil: addent -password -p HTTP/mywebserver.company.com@MYCOMPANY.COM -k 0
-e arcfour-hmac
Password for HTTP/mywebserver.company.com@MYCOMPANY.COM :
ktutil: addent -password -p HTTP/mywebserver.company.com@MYCOMPANY.COM -k 0
-e aes128-cts-hmac-sha1-96
Password for HTTP/mywebserver.company.com@MYCOMPANY.COM :
ktutil: addent -password -p HTTP/mywebserver.company.com@MYCOMPANY.COM -k 0
-e aes256-cts-hmac-sha1-96
Password for HTTP/mywebserver.company.com@MYCOMPANY.COM :
ktutil: wkt HTTP.keytab
ktutil: quit
```

Output 1. Output from Using ktutil to Create a Keytab File

If you are using MIT Kerberos as your KDC and you used the `-randkey` option when adding the principal, you will not know the password or long-term key associated with the principal. The `kadmin` tool can therefore be used to create the Kerberos keytab. Here is an example:

```
kadmin -q "ktadd -k HTTP.keytab HTTP/mywebserver.company.com"
```

Kerberos Configuration Files

Most services will use DNS network queries to discover information about your organization's Kerberos setup. For example, they obtain information about the Kerberos Realm and the location of Kerberos KDCs. However, you might find that it is necessary to provide a Kerberos configuration file to store this information. Quite often this will be required if you either have a complex cross-realm structure (Rogers 2017), or a widely geographically split group of KDCs, in which case you will want the environment to use a specified group of KDCs in the same data center.

For your Linux hosts, it will be easiest to work with the default Kerberos configuration file, located in `/etc/krb5.conf`. If you are using the default file, ensure that all the operating-system accounts can read the contents of this file. Or, for SAS Logon Manager, you can specify a different Kerberos configuration file to be used through Java Virtual Machine (JVM) options. For the CAS server and the SAS Launcher Server, an alternative Kerberos configuration file can be specified with the environment variable `KRB5_CONFIG` (MIT 2018).

MICROSOFT WINDOWS PREREQUISITES

If you have a Microsoft Windows SAS Viya 3.4 deployment, SAS supports only Microsoft Active Directory as your KDC. Your SAS Viya environment will require only two service accounts: one service account for SAS Logon Manager, and one for CAS. The SAS Launcher Server will leverage the computer object where SAS Viya is deployed.

SAS provides a tool to assist you in completing the prerequisites on Microsoft Windows. SAS Viya Deployment Assistant for Windows validates system settings required by SAS Viya on Microsoft Windows hosts. You can access it here:

<https://support.sas.com/en/documentation/install-center/viya/deployment-tools/34/deployment-assistant-windows.html>. With a command-line option, it can also remediate any system settings that do not pass the validation test. More information is available in the [SAS Viya on Windows: Deployment Guide](#).

Service Accounts

SAS Viya running on Microsoft Windows requires two service accounts for Kerberos authentication. The first service account, for SAS Logon Manager, does not require any additional security privileges. This service account will not run any operating-system process within the SAS Viya environment and is used only as part of the authentication process.

The second required service account is for CAS. This service account will run the CAS controller process. SAS recommends using the name "cas" for this account, but the name must be unique in the domain. The maximum length of the name is 20 characters. This service account requires the following privileges on the SAS Viya host:

- Member of the local Administrators group
- Log on as a Service
- Replace Process Level Token

More information about the CAS service account can be found in the [SAS Viya on Windows: Deployment Guide](#).

The SAS Launcher Server does not require a service account because it will run as the local system account on your Windows host.

User Principal Names

In Microsoft Active Directory's implementation of the Kerberos authentication protocol, users are uniquely identified by their User Principal Name. Kerberos credentials can be generated only for a User Principal Name. SAS recommends that the User Principal Name for the SAS Logon Manager and CAS service accounts be the same as the Service Principal Name.

Service Principal Names

Service Principal Names (SPNs) uniquely identify an instance of a service running on a specific host. The SPN has the format *Service-Class/Fully-Qualified-Host-Name@Kerberos-Realm*, where the three service classes are:

- SAS Logon Manager = HTTP
- SAS Cloud Analytic Services = sascas
- SAS Launcher Server = sas-launcher

The fully qualified host name will correspond to the host where the client accesses the service. For most cases this will be the Microsoft Windows host where SAS Viya is deployed. However, for SAS Logon Manager, it could be a separate reverse proxy.

The SPNs for SAS Logon Manager and CAS must be manually registered against the respective service accounts. The Microsoft Windows setspn command-line tool can be used to perform this task. Here is an example:

```
setspn -s HTTP/mywebserver.company.com sas-logon
```

The setspn command requires Write access to the service account in Microsoft Active Directory, which might mean that you need a Domain Administrator to run the commands for you.

The SAS Launcher Server does not require a Service Principal Name to be manually registered. Because the SAS Launcher Server runs as the local system account, it will automatically register the sas-launcher/*host-name* principal against the computer object in Microsoft Active Directory.

Kerberos Keytab Files

For your SAS Viya environment on Windows, a Kerberos keytab file is required only for SAS Logon Manager. The CAS server does not need a Kerberos keytab file because the process is running as the service account. Equally, SAS Launcher Server does not need a Kerberos keytab file because it is running as the local system user.

SAS recommends that the Kerberos keytab file for SAS Logon Manager contain both the User Principal Name and Service Principal Name values, along with all possible encryption types that might be used in Service Tickets presented by client browsers.

As with Linux environments, SAS recommends using the `ktutil` command-line utility to create the Kerberos keytab. However, if you have access only to Microsoft Windows machines, you will not be able to use this utility. A Microsoft-provided tool can be used instead, but this tool can make direct changes to the service account in Microsoft Active Directory. The following example command creates a Kerberos keytab and attempts to prevent the tool from changing values in Microsoft Active Directory:

```
ktpass /out HTTP.keytab /princ HTTP/mywebserver.company.com@MYCOMPANY.COM -  
SetUPN /mapuser sas-logon /mapop set /crypto all /ptype KRB5_NT_PRINCIPAL  
/pass Password -SetPass
```

The option `-SetUPN` prevents `ktpass` from changing the User Principal Name, and the option `-SetPass` prevents `ktpass` from changing the service account password. The option `/mapop set` replaces the current SPN value with the value of `/princ`. Because you are updating the SPN, you need to have Write access to the service account, which normally means that you need Domain Administrator privileges.

The `ktpass` tool will be available on domain controllers or other hosts where Remote Server Administration Tools (RSTAT) has been installed.

Kerberos Configuration Files

SAS Viya on Windows includes a Kerberos configuration file for SAS Logon Manager. CAS and SAS Launcher Server do not require a Kerberos configuration file. The Kerberos configuration file for SAS Logon Manager is installed in the following location:

```
C:\ProgramData\SAS\Viya\etc\sysconfig\sas-javaesntl\krb5.ini
```

This file is automatically included in the SAS Logon Manager start-up and contains the following content:

```
[libdefaults]  
forwardable = true
```

If you need to add content, perhaps to define cross-realm structures, you can update the Kerberos configuration file and restart the SAS Logon Manager Windows Service.

GENERAL PREREQUISITES

In addition to the operating system-specific prerequisites that are detailed above, some general prerequisites must also be completed.

Delegation

As explained in the introduction and Kerberos overview sections, the aim of this configuration is for Kerberos authentication to flow into, between, and out from the SAS Viya environment. In Kerberos terms, this is referred to as *delegation*. To enable end users to authenticate to SAS Logon Manager and to the CAS server and SAS Launcher Server

using their identities and Kerberos, you must also enable delegation. Equally, delegation is required for the authentication flow to continue from either CAS or the SAS Compute Server session to Hadoop.

Kerberos delegation must be enabled for three services: SAS Logon Manager, CAS, and SAS Launcher Server. If you are using MIT Kerberos as your KDC, then enabling delegation involves setting the flag `ok_as_delegate` on the principal. For example, the following command will add this flag to the existing HTTP principal:

```
kadmin -q "modprinc +ok_as_delegate HTTP/mywebserver.company.com"
```

More information about the `kadmin` tool is available in the MIT Kerberos documentation (MIT 2018).

If you are using Microsoft Active Directory for your KDC, you must set the delegation option after registering the SPN. The Active Directory Users and Computers GUI tool will not expose the delegation options until at least one SPN is registered against the service account or computer object.

SAS Viya 3.4 supports only unconstrained delegation. For service accounts, you must select the option to **Trust this user for delegation to any service (Kerberos only)**, as shown in Figure 6. By contrast, for the SAS Launcher Server on Microsoft Windows, it must be against the computer object, and the option is **Trust this computer for delegation to any service (Kerberos only)**.

Because SAS Viya 3.4 supports only unconstrained delegation, Microsoft Windows Defender Credential Guard cannot be enabled when you use Kerberos delegation. SAS plans to enhance SAS Viya to support constrained delegation in the future so that Microsoft Windows Defender Credential Guard can be supported.

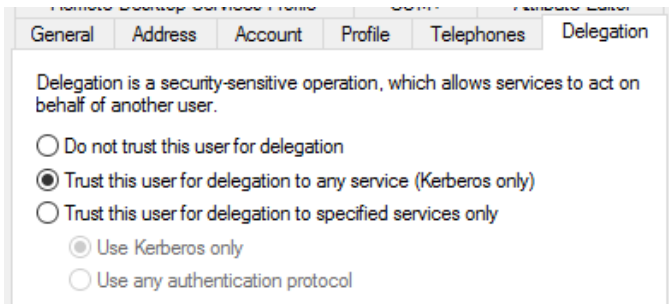


Figure 6. Microsoft Active Directory Delegation User Option

SAS Administrators

The final prerequisite to consider is your administrative users. Once Kerberos authentication is enabled for SAS Logon Manager, no other authentication mechanism can be used to access the SAS Viya visual interfaces, such as SAS Environment Manager. You must ensure that members of your SAS Administrators group are able to correctly log in before you enable Kerberos authentication.

Client Browsers

The client browsers that your end users will use to connect to your SAS Viya environment must also be configured for Kerberos delegation. The configuration settings are different for Microsoft Internet Explorer, Google Chrome, and Mozilla Firefox. The SAS Viya Administration documentation includes a section on [Microsoft Internet Explorer and Google Chrome](#) as well as [Mozilla Firefox](#). You will need to complete these configuration steps on all your end users' client machines.

CONFIGURATION

As with the prerequisites that we examined in the previous section the configuration is different depending on the operating system of the SAS Viya 3.4 environment. In this section, we will first address the steps to configure Kerberos authentication on Linux and then on Microsoft Windows.

LINUX CONFIGURATION

The configuration of the three services on Linux environments is completed in three different ways. First, many of the configuration options can be set using SAS Environment Manager. Then some settings must be changed in configuration files for CAS. Finally, with SAS Viya 3.4, one option must be set using a command-line tool.

SAS Logon Manager

Kerberos configuration for SAS Logon Manager is completed in SAS Environment Manager by a member of the SAS Administrators group. There are no configuration files to edit, with the possible exception of the Kerberos configuration file. In SAS Environment Manager, select **Configuration, Definitions** and create a new configuration for **sas.logon.kerberos**. The properties are described in Table 1:

Field	Value
debug	On – Causes debug messages to be logged.
holdOnToGSSContext	On – Required to enable Kerberos delegation from SAS Logon Manager.
keyTabLocation	Uniform Resource Identifier (URI). For example, <code>file:///opt/sas/http.keytab</code>
servicePrincipal	Principal Name from keytab. Note: If the environment includes multiple realms, this should include the realm. Examples: <code>HTTP/fully.qualified.hostname@REALM</code> or <code>user@REALM</code>
spn	The HTTP SPN, if different from the principal name in the keytab. Note: If the environment includes multiple realms, this should include the realm. An example would be <code>HTTP/fully.qualified.hostname@REALM</code>
stripRealmForGss	On – Causes realm to be removed from User Principal.

Table 1. SAS Logon Manager Kerberos Configuration Options

These configuration options in SAS Environment Manager are shown in Display 1.

The setting `holdOnToGSSContext` will cause SAS Logon Manager to store the delegated Kerberos credentials with the Credentials microservice. This will make the delegated Kerberos credentials available for out-bound connections from the visual interfaces.

The setting `stripRealmForGss` will be on by default. This setting causes SAS Logon Manager to remove the Kerberos realm from the user name obtained from the Kerberos Service Ticket. Thus `username@REALM` becomes `username` for the search in the Identities microservice to return user and group information. This can be disabled if you need to perform this search on the full `username@REALM` value.

Edit sas.logon.kerberos Configuration

debug: Enables the debug mode of the JAAS Kerberos login module.

holdOnToGSSContext: Enable Kerberos credentials to be delegated to other services.

keyTabLocation: The URI of the keytab file.

servicePrincipal: The name of the service principal in the keytab.

spn: The HTTP service principal name (SPN), if different than the principal name in the keytab.

stripRealmForGss: When enabled, this option strips the realm from the user name.


Display 1. The sas.logon.kerberos Configuration Options in SAS Environment Manager

In addition to defining the properties for **sas.logon.kerberos**, it is also necessary to set Kerberos as active. This is also completed in SAS Environment Manager. Select **Configuration, All Services, SAS Logon Manager**, and then update the settings for **spring**. The property **profiles.active** must have *kerberos* added to the comma-separated list. The final setting is shown in Display 2.

Edit spring Configuration

GUID: f360467d-6f1b-4cdc-9417-b71109d889e9
The globally unique identifier for the configuration instance.

Services: One or more services to which this configuration instance applies.

profiles.active: 

[+ Add property](#)

Display 2. SAS Logon Manager spring Configuration.

If a Kerberos configuration file needs to be specified because it is not in the default location, this task can also be completed in SAS Environment Manager. Select **Configuration, All Services, SAS Logon Manager**, and then update the settings for **jvm**. An additional property can be created to point to the Kerberos configuration file, as shown in Display 3.

Display 3. SAS Logon Manager Specifying Alternative Kerberos Configuration File

This completes the required configuration changes for SAS Logon Manager, and the process can be restarted:

```
systemctl restart sas-viya-saslogon-default
```

The CAS Server

The configuration for Kerberos authentication is completed in two locations: configuration files and SAS Environment Manager. Both areas must be completed in order to configure Kerberos authentication for all types of use.

The configuration file settings for Kerberos can be manually entered into the following file:

```
/opt/sas/viya/config/etc/cas/default/casconfig_usermods.lua
```

Or you can place these settings in the vars.yml file and run the deployment playbook again. When they are made using vars.yml, the settings automatically appear in the following file:

```
/opt/sas/viya/config/etc/cas/default/casconfig_deployment.lua
```

Take these steps to configure the settings in casconfig_usermods.lua or in vars.yml:

1. Update the `cas.provlist` to include `kerb`:
 - In `casconfig_usermods.lua`, this setting should be:
`cas.provlist = 'oauth.ext.kerb'`
 - In `vars.yml`, this setting will be under the `cfg:` section and should be:
`provlist: 'oauth.ext.kerb'`
2. (Optional) Set the server principal:
 - In `casconfig_usermods.lua`, this setting should be:
`env.CAS_SERVER_PRINCIPAL = 'CAS/HOSTNAME.COMPANY.COM'`
 - In `vars.yml`, this setting will be under the `env:` section and should be:
`CAS_SERVER_PRINCIPAL: 'CAS/HOSTNAME.COMPANY.COM'`

3. (Optional) Set the Kerberos keytab location:

- In `casconfig_usermods.lua`, this setting should be:
`env.KRB5_KTNAME = '/opt/sas/cas.keytab'`
- In `vars.yml`, this setting will be under the `env:` section and should be:
`KRB5_KTNAME: '/opt/sas/cas.keytab'`


It is necessary to make these changes to the configuration files to enable Kerberos authentication from CAS. But these changes are not sufficient to also allow Kerberos authentication into CAS from visual interfaces.

Two additional tasks must be completed by a member of the SAS Administrators group in SAS Environment Manager in order to complete the configuration of Kerberos for CAS. First, a configuration option must be specified that informs CAS that Kerberos authentication is configured. The option **kerberos.enabled** must be set to true. This can be found in the **Configuration** section of SAS Environment Manager under **Definitions**, and under **sas.compute** as shown in Display 4. This configuration setting is shared with the SAS Compute Server.

The second task to complete in SAS Environment Manager is the creation of the custom group. For CAS to attempt to run the session as the end user, the end user must be a member of a custom group. The name of the custom group is not important and can be anything. The ID of the custom group must be `CASHostAccountRequired`. Without membership in this group, the CAS session will be launched as the operating system account running the CAS controller. By default, this will be the "cas" account, but this can be changed at deployment time.

Edit sas.compute Configuration

GUID: 4cf16daf-11ed-458f-841a-5774f12a7b30
The globally unique identifier for the configuration instance.

Services: Global 
Services to which this configuration instance applies. 'Global' indicates the configuration instance applies to all services.

domain.default: DefaultAuth
The default authentication domain to use for looking up host credentials.

kerberos.enabled:
Authenticate to compute servers using Kerberos.

serviceAccount.default:
The default service account that should be used to run jobs on the host.

Display 4. The `sas.compute` Configuration Options in SAS Environment Manager

This completes the configuration changes for CAS. The process for the CAS Controller must be restarted to pick up the changes:

```
systemctl restart sas-viya-cascontroller-default
```

SAS Launcher Server

There are two steps to complete the configuration of Kerberos authentication for the SAS Launcher Server. First, the configuration property in SAS Environment Manager must be set. This property is shared with CAS, so if CAS has been configured for Kerberos, this will have already been completed.

In SAS Environment Manager, as a member of SAS Administrators, select **Configuration, Definitions**, and select **sas.compute**. Update the configuration and set **kerberos.enabled** to true, as shown in Display 4. This is required for the SAS Launcher Server to leverage Kerberos authentication.

Next, a configuration option must be set that points to the Kerberos keytab for the SAS Launcher Server. With SAS Viya 3.4, you cannot set this within SAS Environment Manager. The property must be set in the SAS Configuration Server using the SAS Bootstrap Config tool. To use the SAS Bootstrap Config tool, several environment variables must be set in the current session. To set the environment variables, run the following two commands:

```
source /opt/sas/viya/config/consul.conf
export CONSUL_TOKEN=`cat
/opt/sas/viya/config/etc/SASSecurityCertificateFramework/tokens/consul/
default/client.token`
```

With the environment variables set, the SAS Bootstrap Config tool can be used to set the configuration property:

```
/opt/sas/viya/home/bin/sas-bootstrap-config kv write --force --key
config/launcher-server/global/keytab --value /path/to/keytab.file
```

The *path/to/keytab.file* is the full path to the Kerberos keytab file.

This completes the configuration of the SAS Launcher Server for Kerberos authentication, and the process can be restarted:

```
systemctl restart sas-viya-runlauncher-default
```

MICROSOFT WINDOWS CONFIGURATION

Kerberos authentication is the only supported authentication mechanism for SAS Viya 3.4 on Microsoft Windows. However, minimal configuration is required to enable Kerberos authentication on Windows. Only the SAS Logon Manager needs to be configured.

SAS Logon Manager

The configuration for SAS Logon Manager in Microsoft Windows SAS Viya environments is virtually the same as for Linux, with one key difference. The main difference for Microsoft Windows is how you specify the location of the Kerberos keytab. In SAS Environment Manager, you enter a URI to point to the file. For Linux, this was entered as `file:///opt/sas/http.keytab`. For Windows, you must enter `file:///d:/sas/HTTP.keytab`.

This points to a file, `d:\sas\HTTP.keytab`, which can be placed anywhere on the file system as long as the local service account has access to it. Notice that backslashes are used, and the colon for the drive letter is still required. Also, any spaces in the pathname must be URI encoded as `%20`.

Otherwise, the configuration is the same.

CAS Server

With a Microsoft Windows SAS Viya environment, no additional configuration is required for CAS to use Kerberos authentication.

SAS Launcher Server

With a Microsoft Windows SAS Viya environment, no additional configuration is required for the SAS Launcher Server to use Kerberos authentication.

EDGE CASES

SCHEDULING

Within an environment that is configured for Kerberos authentication, scheduling presents challenges. By design, Kerberos tickets are valid only for relatively short periods of time. Kerberos defines two lifetimes for a given ticket: the current lifetime of the ticket, and, if configured with the KDC, a renewable lifetime. A ticket that is still within its initial lifetime can be renewed up to its renewable lifetime. Once the renewable lifetime has expired, the end user must authenticate again. Not all tickets are renewable, which means that they expire as soon as the ticket lifetime expires.

If we consider Microsoft Active Directory, by default your user's Kerberos TGT will be valid for 10 hours and can be renewed for 7 days. So if you log in to your SAS Viya environment at 9:00 am and schedule a task to run every day at 9:00 pm, you should expect some issues. Even the first run of the task is going to take place 12 hours after you logged in, so your TGT will have expired before the task is run.

SAS provides a couple of technologies to assist with the expiring credentials. First, as long as you remain logged in to one of the SAS Viya visual interfaces and are active in the application, your Kerberos credentials are refreshed. The keepalive messages that are triggered by the different visual interfaces, such as SAS Environment Manager or SAS Visual Analytics, require re-authentication to SAS Logon Manager. This means that the stored TGT is refreshed if you are logged in. So, in our example above, if you remain logged in until 5:00 pm, your TGT will now be valid for running that task at 9:00 pm.

But what about the weekends, or days when you take a vacation? SAS provides another technology: the ability for CAS and the SAS Launcher Server to fall back to using a user name and password. Users are not prompted for this user name and password; the credentials must be stored beforehand. The credentials are stored in SAS Environment Manager in an Authentication Domain named DefaultAuth by default. The user name and password can be stored individually for a user, or they can be stored for a group and made available to all members of the group. Only one user name and password set should be available to a given user. This is the authentication flow that we illustrated in Figure 3. For SAS Viya deployments on Linux, you must ensure that the PAM stack correctly generates a Kerberos credentials cache for these users.

Finally, for a SAS Viya environment on Linux, the CAS server can also fall back to directly launching the session. With such a session, users who are members of CASHostAccountRequired will not have access to any Kerberos credentials. As a result, they cannot access your secured third-party data source. However, this session will be able to access data already loaded into memory, which is useful for running scheduled reports.

Therefore, you can see that scheduling adds complexity to the Kerberos configuration, but does not become impossible. You just need to carefully plan who will be able to schedule, depending on the resources that they need to access.

SAS ADMINISTRATION CLI AND SAS VISUAL ANALYTICS APPS

Both SAS Viya 3.4 on Windows and the late-2018 release of SAS Viya 3.4 on Linux support Kerberos for the SAS Administration Command-Line Interface (CLI) and the SAS Visual Analytics Apps on mobile devices. Even when SAS Logon Manager is configured for Kerberos authentication, both applications still allow for connections using user name and password. This is because these applications access a different end-point of SAS Logon Manager that is not secured with Kerberos. However, when a connection is made using a user name and password, SAS Logon Manager initializes a Kerberos credential for your end-user. This Kerberos credential is then stored with the Credentials microservice and is available for additional connections, such as to CAS.

In addition, the latest version of the SAS Administration CLI can also use Kerberos authentication for logging in to your SAS Viya 3.4 environment. You can download it here: <https://support.sas.com/downloads/package.htm?pid=2133>. Kerberos support for logging in with the SAS Administration CLI differs based on the execution platform (Windows, Linux, OS-X):

- The Windows support uses the Windows SSPI via system calls.
- The Linux support uses a GSSAPI implementation using CGO to call the underlying GSSAPI C libraries.
- The direct OS-X support is currently disabled.

If you plan to run the SAS Administration CLI from a Linux host, the correct underlying GSSAPI C libraries must be installed. On a Red Hat Linux system, this can be completed with the following command, running as root:

```
yum install -y krb5-libs
```

This command will provide the `/usr/lib64/libgssapi_krb5.so.2` library. You will then need to create a symbolic link, `/usr/lib64/libgssapi_krb5.so`, which points to the `libgssapi_krb5.so.2` library. You can add the symbolic link by running the following command as root:

```
ln -s /usr/lib64/libgssapi_krb5.so.2 /usr/lib64/libgssapi_krb5.so; ldconfig
```

As an alternative, you could install the `krb5-devel` package, which includes this symbolic link. However, this package will install additional items that are not required. On a SUSE Linux system, this can be installed by running the following command as root:

```
zypper install krb5
```

This provides the necessary infrastructure for the SAS Administration CLI to leverage Kerberos for authentication. Before using the SAS Administration CLI that you have [downloaded](#), you need to install the CLI plug-ins and set up the profile. These steps are covered in detail in the [documentation](#). Even if using the SAS Administration CLI on a Linux host where the original version has been run, you must still set up the profile again.

Using Kerberos authentication to log in to the SAS Administration CLI requires that you change the initial command to log in to the SAS Viya environment. Run the following command:

```
./sas-admin auth kerberos
```

You should see the following message:

```
Login succeeded. Token saved.
```

If you include `--verbose` in the command, you can review the HTTP requests/responses sent to SAS Logon Manager. You will then see the 401 HTTP response from the request to SAS Logon Manager with the `Www-Authenticate: Negotiate` header. Notice that this GET request is to `/SASLogon`. As an alternative, when using user name and password, the command is as follows:

```
./sas-admin auth login
```

If you include `--verbose`, the command now shows a POST message sent to `/SASLogon/oauth/token` after entering your user name and password.

CONCLUSION

The use of Kerberos authentication across organizations is becoming more common. This paper has described how SAS Viya 3.4 can use Kerberos authentication, giving you a better understanding of how Kerberos is used throughout your environment. We focused heavily on the prerequisites that you must ensure are completed correctly. Most issues with Kerberos configuration can be traced back to issues with completing the prerequisites. We then walked through the configuration steps that you must complete within your environment. I hope that this paper has provided the information that you need to get this correctly implemented.

SAS has been on a journey with our customers, making the implementation of Kerberos more straightforward and painless since SAS 9.2, through both SAS 9.3 and SAS 9.4, and now on to SAS Viya. The configuration steps that are required for SAS Viya 3.4 are now much simpler than with previous releases of SAS software.

I hope that you find this paper a valuable resource. Additional documentation is provided in the following references.

REFERENCES

- Massachusetts Institute of Technology (MIT). "MIT Kerberos Documentation." Available <http://web.mit.edu/Kerberos/krb5-latest/doc/index.html>. Accessed on January 7, 2019.
- Roda, Mike. 2016. "Tips and Best Practices for Configuring Integrated Windows Authentication." *Proceedings of the SAS Global Forum 2016 Conference*. Cary, NC: SAS Institute Inc. Available: <http://support.sas.com/resources/papers/proceedings16/SAS3720-2016.pdf>.
- Rogers, Stuart J. 2017. "Kerberos Cross-Realm Authentication: Unraveling the Mysteries." *Proceedings of the SAS Global Forum 2017 Conference*. Cary, NC: SAS Institute Inc. Available: <http://support.sas.com/resources/papers/proceedings17/SAS0623-2017.pdf>.
- Rogers, Stuart J. 2016. "Kerberos Delegation with SAS® 9.4." *Proceedings of the SAS Global Forum 2016 Conference*. Cary, NC: SAS Institute Inc. Available: <http://support.sas.com/resources/papers/proceedings16/SAS3443-2016.pdf>.
- Rogers, Stuart J. 2017. "Kerberos Cross-Realm Authentication: Unraveling the Mysteries." *Proceedings of the SAS Global Forum 2017 Conference*. Cary, NC: SAS Institute Inc. Available: <http://support.sas.com/resources/papers/proceedings17/SAS0623-2017.pdf>.
- Rogers, Stuart J. 2018. "SAS® 9.4 on Windows®: Unleashing Kerberos on Hadoop." *Proceedings of the SAS Global Forum 2018 Conference*. Cary, NC: SAS Institute Inc. Available: <https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/1878-2018.pdf>.

ACKNOWLEDGMENTS

I would like to thank the following people for taking the time to review and contribute to this paper:

- Chuck Hunley
- Larry Noe
- Mike Roda

RECOMMENDED READING

- SAS Viya 3.4 Administration
- SAS 9.4 and SAS Viya 3.4 Programming Documentation
- SAS Viya 3.4 on Windows: Deployment Guide

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Stuart J Rogers
SAS Institute Inc.
stuart.rogers@sas.com
<http://www.sas.com>

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.