Lasso - Development #54037

SHA-1 usage in lasso

17 mai 2021 16:44 - Jakub Hrozek

Statut:	Fermé	**Début:**	17 mai 2021
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Non	**Planning:**	Non

Description

RHEL-9 is trying to get rid of any usages of SHA-1 across the codebase. Since I maintain lasso and mod_auth_mellon, I was looking at how are the different digests used in lasso and wanted to get your advice since I'm not all that familiar with lasso code.

Here's what I was thinking on a high-level:
- add a configure-time switch --enable-sha1. upstream it would default to enabled, RHEL-9 would disable it.
- based on the switch value, there would be a default signature method
- add a function lasso_allowed_signature_method along the lines of lasso_validate_signature_method that would return always true if sha1 signatures are allowed and if sha1 is not allowed in the distribution, return true if the signature uses something else than sha1
- this would be used in lasso_node_impl_init_from_xml() and in all the functions in lasso/xml/tools.c like lasso_query_sign() lasso_query_verify_helper() or _lasso_xmlsec_load_key_from_buffer() or lasso_xmlnode_add_saml2_signature_template()
- there's a bunch of SHA1 based digests that are not configurable all over the place under lasso/id-ff/. I was thinking about just mass converting them to the default signature method, so nothing would change unless you configure lasso with --disable-sha1
- I don't know what to do about the WSF subtree at all. I don't know anything about WSF, but I was thinking doing the above there as well.

Note that I'm fine working on the code changes, but before I do that, I wanted to hear your opinion so that I don't work on something that would be rejected later.

What do you think about the above?

Révisions associées

Révision 8b8fd22a - 23 juin 2021 23:32 - Jakub Hrozek

Fix lasso_query_sign HMAC other than SHA1 (#54037)

The switch clause was using SHA1 digests for all digest types when
signing. This obviously breaks verifying the signatures if HMAC-SHAXXX
is used and XXX is something else than 1.

Révision f625eaa0 - 23 juin 2021 23:32 - Jakub Hrozek

tests: Move test08_lasso_key and test07_saml2_query_verify_signature to SHA256 (#54037)

These tests use a hardcoded query and private key which makes it
unsuitable to make the tests use the configured default digest. Let's
just convert them to SHA256 unconditionally.

Révision f095ac8f - 23 juin 2021 23:32 - Jakub Hrozek

Make the default signature method and the minimal hash strength configurable (#54037)

Adds two new configure options:
--with-default-sign-algo
--min-hash-algo

--with-default-sign-algo sets the default signing algorithm and defaults
to rsa-sha1. At the moment, two algorithms are supported: rsa-sha1 and
rsa-sha256.

--min-hash-algo sets the minimum hash algorithm to be accepted. The
default is sha1 for backwards compatibility as well.

Related:
https://dev.entrouvert.org/issues/54037

Révision 0d34c97b - 23 juin 2021 23:32 - Jakub Hrozek

Mass-replace LASSO_SIGNATURE_METHOD_RSA_SHA1 with lasso_get_default_signature_method() (#54037)

This should be backwards-compatible but at the same time use the
selected default instead of RSA-SHA1.

Related:
https://dev.entrouvert.org/issues/54037

Révision f9a3aca0 - 24 juin 2021 02:15 - Jakub Hrozek

Check if the signature method is allowed in addition to being valid (#54037)

Adds a new utility function lasso_allowed_signature_method() that checks
if the signature method is allowed. Previously, the code would only
check if the method was valid.

This new function is used whenever lasso_validate_signature_method was
previously used through lasso_ok_signature_method() which wraps both
validate and allowed.

lasso_allowed_signature_method() is also used on a couple of places,
notably lasso_query_verify_helper().

Related:
https://dev.entrouvert.org/issues/54037

Révision f70eee9e - 24 juin 2021 02:15 - Jakub Hrozek

python: Skip the DSA key test unless SHA-1 is configured (#54037)

lasso supports DSA-XXX only with SHA-1. The alternative is to use
DSA-SHA256.

Révision 1b0000e0 - 24 juin 2021 02:15 - Jakub Hrozek

test13_test_lasso_server_load_metadata: Don't verify signature if lasso is not configured with sha-1 (#54037)

Historique

#1 - 17 mai 2021 22:04 - Benjamin Dauvergne

Jakub Hrozek a écrit :

> RHEL-9 is trying to get rid of any usages of SHA-1 across the codebase. Since I maintain lasso and mod_auth_mellon, I was looking at how are the different digests used in lasso and wanted to get your advice since I'm not all that familiar with lasso code.
>
> Here's what I was thinking on a high-level:
> - add a configure-time switch --enable-sha1. upstream it would default to enabled, RHEL-9 would disable it.
> - based on the switch value, there would be a default signature method
> - add a function lasso_allowed_signature_method along the lines of lasso_validate_signature_method that would return always true if sha1 signatures are allowed and if sha1 is not allowed in the distribution, return true if the signature uses something else than sha1
> - this would be used in lasso_node_impl_init_from_xml() and in all the functions in lasso/xml/tools.c like lasso_query_sign() lasso_query_verify_helper() or _lasso_xmlsec_load_key_from_buffer() or lasso_xmlnode_add_saml2_signature_template()
> - there's a bunch of SHA1 based digests that are not configurable all over the place under lasso/id-ff/. I was thinking about just mass converting them to the default signature method, so nothing would change unless you configure lasso with --disable-sha1

It looks fine to me.

> - I don't know what to do about the WSF subtree at all. I don't know anything about WSF, but I was thinking doing the above there as well.

You can ignore it, it's more than deprecated, just use --disable-wsf in your builds; saying that I just tested it and some #ifdef in C code and Makefile are lacking for it to work immediately. Restoring --disable-wsg still seems simpler than trying to fix this dead code.

#2 - 16 juin 2021 14:17 - Jakub Hrozek

On Mon, May 17, 2021 at 10:04:04PM +0200, redmine@entrouvert.com wrote:

> Issue #54037 has been updated by Benjamin Dauvergne.
>
> Jakub Hrozek a écrit :
>
> > RHEL-9 is trying to get rid of any usages of SHA-1 across the codebase. Since I maintain lasso and mod_auth_mellon, I was looking at how

> > are the different digests used in lasso and wanted to get your advice since I'm not all that familiar with lasso code.
> >
> > Here's what I was thinking on a high-level:
> > - add a configure-time switch --enable-sha1. upstream it would default to enabled, RHEL-9 would disable it.
> > - based on the switch value, there would be a default signature method
> > - add a function lasso_allowed_signature_method along the lines of lasso_validate_signature_method that would return always true if sha1 signatures are allowed and if sha1 is not allowed in the distribution, return true if the signature uses something else than sha1
> > - this would be used in lasso_node_impl_init_from_xml() and in all the functions in lasso/xml/tools.c like lasso_query_sign() lasso_query_verify_helper() or _lasso_xmlsec_load_key_from_buffer() or lasso_xmlnode_add_saml2_signature_template()
> > - there's a bunch of SHA1 based digests that are not configurable all over the place under lasso/id-ff/. I was thinking about just mass converting them to the default signature method, so nothing would change unless you configure lasso with --disable-sha1
>
> It looks fine to me.

Great, what do you think about the attached patch?

> > - I don't know what to do about the WSF subtree at all. I don't know anything about WSF, but I was thinking doing the above there as well.
>
> You can ignore it, it's more than deprecated, just use --disable-wsf in your builds; saying that I just tested it and some #ifdef in C code and Makefile are lacking for it to work immediately. Restoring --disable-wsg still seems simpler than trying to fix this dead code.

Actually, turns out that lasso in Fedora and RHEL already disables wsf,
cool.

--x5j2zftuhtf6h6h7
Content-Type: text/plain; charset=us-ascii
Content-Disposition: attachment;
filename="0001-Fix-lasso_query_sign-HMAC-other-than-SHA1.patch"

From 84ad9615223f84d56e486a26a3cf9906c4cb5a5d Mon Sep 17 00:00:00 2001
From: Jakub Hrozek <jhrozek@redhat.com>
Date: Wed, 16 Jun 2021 10:18:30 +0200
Subject: [PATCH 1/6] Fix lasso_query_sign HMAC other than SHA1

The switch clause was using SHA1 digests for all digest types when
signing. This obviously breaks verifying the signatures if HMAC-SHAXXX
is used and XXX is something else than 1.

```
---
lasso/xml/tools.c         | 35 +++++++++++++++++++++++++-----------
tests/login_tests_saml2.c |  6 ++---
2 files changed, 26 insertions(), 15 deletions(-)

diff --git a/lasso/xml/tools.c b/lasso/xml/tools.c
index 96d88a2c4..290fd55f2 100644
--- a/lasso/xml/tools.c
++ b/lasso/xml/tools.c
@ -594,22 +594,20 @ lasso_query_sign(char query, LassoSignatureContext context)
sigret_size = DSA_size(dsa);
break;
case LASSO_SIGNATURE_METHOD_HMAC_SHA1:
md = EVP_sha1();
+           sigret_size = EVP_MD_size(md);
+           break;
case LASSO_SIGNATURE_METHOD_HMAC_SHA256:
+           md = EVP_sha256();
+           sigret_size = EVP_MD_size(md);
+           break;
case LASSO_SIGNATURE_METHOD_HMAC_SHA384:
+           md = EVP_sha384();
+           sigret_size = EVP_MD_size(md);
+           break;
case LASSO_SIGNATURE_METHOD_HMAC_SHA512:
-           if ((rc = lasso_get_hmac_key(key, (void*)&hmac_key,
-                                       &hmac_key_length))) {
-               message(G_LOG_LEVEL_CRITICAL, "Failed to get hmac key (%s)", lasso_strerror(rc));
-               goto done;
-           }
-           g_assert(hmac_key);
-           md = EVP_sha1();
+           md = EVP_sha512();
sigret_size = EVP_MD_size(md);
```

```
-          /* key should be at least 128 bits long /
-          if (hmac_key_length < 16) {
-              critical("HMAC key should be at least 128 bits long");
-              goto done;
-          }
break;
default:
g_assert_not_reached();
@ -645,6 +643,19 @ lasso_query_sign(char *query, LassoSignatureContext context)
case LASSO_SIGNATURE_METHOD_HMAC_SHA256:
case LASSO_SIGNATURE_METHOD_HMAC_SHA384:
case LASSO_SIGNATURE_METHOD_HMAC_SHA512:
+          if ((rc = lasso_get_hmac_key(key, (void*)&hmac_key,
+                                       &hmac_key_length))) {
+              message(G_LOG_LEVEL_CRITICAL, "Failed to get hmac key (%s)", lasso_strerror(rc));
+              goto done;
+          }
+          g_assert(hmac_key);

/* key should be at least 128 bits long */
+          if (hmac_key_length < 16) {
+              critical("HMAC key should be at least 128 bits long");
+              goto done;
+          }

HMACnew_query,
strlen(new_query), sigret, &siglen);
status = 1;
diff --git a/tests/login_tests_saml2.c b/tests/login_tests_saml2.c
index e331c07a7..e1d78b5b1 100644
--- a/tests/login_tests_saml2.c
++ b/tests/login_tests_saml2.c
@ -981,7 +981,7 @ sso_initiated_by_sp(LassoServer *idp_context, LassoServer *sp_context, SsoCallba
lasso_release_gobject(sp_login_context);
}

-START_TEST(test07_sso_sp_with_hmac_sha1_signatures)
+START_TEST(test07_sso_sp_with_hmac_sha256_signatures) {
LassoServer *idp_context = NULL;
LassoServer *sp_context = NULL;
@ -990,7 +990,7 @ START_TEST(test07_sso_sp_with_hmac_sha1_signatures)
```

```
/* Create the shared key */
    key = lasso_key_new_for_signature_from_memory("xxxxxxxxxxxxxxxx", 16,
-            NULL, LASSO_SIGNATURE_METHOD_HMAC_SHA1, NULL);
+            NULL, LASSO_SIGNATURE_METHOD_HMAC_SHA256, NULL);
    check_true(LASSO_IS_KEY(key));

/* Create an IdP context for IdP initiated SSO with provider metadata 1 */
@ -1640,7 +1640,7 @ login_saml2_suite()
    tcase_add_test(tc_spSloSoap, test04_sso_then_slo_soap);
    tcase_add_test(tc_idpKeyRollover, test05_sso_idp_with_key_rollover);
    tcase_add_test(tc_spKeyRollover, test06_sso_sp_with_key_rollover);
-    tcase_add_test(tc_hmacSignature, test07_sso_sp_with_hmac_sha1_signatures);
+    tcase_add_test(tc_hmacSignature, test07_sso_sp_with_hmac_sha256_signatures);
    tcase_add_test(tc_spLogin, test08_test_authnrequest_flags);
    tcase_add_test(tc_ecp, test09_ecp);
    tcase_add_test(tc_ecp, test10_ecp);
```

#3 - 16 juin 2021 14:19 - Jakub Hrozek

- Fichier 0001-Fix-lasso_query_sign-HMAC-other-than-SHA1.patch ajouté

- Fichier 0002-tests-Move-test08_lasso_key-and-test07_saml2_query_v.patch ajouté

- Fichier 0003-Make-the-default-signature-method-and-the-minimal-ha.patch ajouté

- Fichier 0004-Mass-replace-LASSO_SIGNATURE_METHOD_RSA_SHA1-with-la.patch ajouté

- Fichier 0005-Check-if-the-signature-method-is-allowed-in-addition.patch ajouté

- Fichier 0006-python-Skip-the-DSA-key-test-unless-SHA-1-is-configu.patch ajouté

Oops, sorry, sending patches to redmine doesn't go so well :-)

I'll attach the patches instead.

#4 - 22 juin 2021 11:02 - Jakub Hrozek

- Fichier 1000-Fix-lasso_query_sign-HMAC-other-than-SHA1.patch ajouté

- Fichier 1001-tests-Move-test08_lasso_key-and-test07_saml2_query_v.patch ajouté

- Fichier 1002-Make-the-default-signature-method-and-the-minimal-ha.patch ajouté

- Fichier 1003-Mass-replace-LASSO_SIGNATURE_METHOD_RSA_SHA1-with-la.patch ajouté

- Fichier 1004-Check-if-the-signature-method-is-allowed-in-addition.patch ajouté

- Fichier 1005-python-Skip-the-DSA-key-test-unless-SHA-1-is-configu.patch ajouté

- Fichier 1006-test13_test_lasso_server_load_metadata-Don-t-verify-.patch ajouté

Submitting a new version of the patches..this time I made sure that lasso works fine even with underlying xmlsec library compiled without sha1 support.

#5 - 22 juin 2021 11:04 - Jakub Hrozek

btw if pulling git trees is easier, see https://github.com/jhrozek/lasso branch sha1

#6 - 24 juin 2021 02:17 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#7 - 24 juin 2021 02:17 - Benjamin Dauvergne

- Statut changé de Nouveau à Résolu (à déployer)

#8 - 04 septembre 2021 10:05 - Benjamin Dauvergne

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-Fix-lasso_query_sign-HMAC-other-than-SHA1.patch	3,78 ko	16 juin 2021	Jakub Hrozek
0003-Make-the-default-signature-method-and-the-minimal-ha.patch	9,56 ko	16 juin 2021	Jakub Hrozek
0002-tests-Move-test08_lasso_key-and-test07_saml2_query_v.patch	13,5 ko	16 juin 2021	Jakub Hrozek
0004-Mass-replace-LASSO_SIGNATURE_METHOD_RSA_SHA1-with-la.patch	7,15 ko	16 juin 2021	Jakub Hrozek
0005-Check-if-the-signature-method-is-allowed-in-addition.patch	6,51 ko	16 juin 2021	Jakub Hrozek
0006-python-Skip-the-DSA-key-test-unless-SHA-1-is-configu.patch	1,14 ko	16 juin 2021	Jakub Hrozek
1000-Fix-lasso_query_sign-HMAC-other-than-SHA1.patch	3,79 ko	22 juin 2021	Jakub Hrozek
1001-tests-Move-test08_lasso_key-and-test07_saml2_query_v.patch	17 ko	22 juin 2021	Jakub Hrozek
1002-Make-the-default-signature-method-and-the-minimal-ha.patch	14,4 ko	22 juin 2021	Jakub Hrozek
1003-Mass-replace-LASSO_SIGNATURE_METHOD_RSA_SHA1-with-la.patch	7,16 ko	22 juin 2021	Jakub Hrozek
1004-Check-if-the-signature-method-is-allowed-in-addition.patch	6,52 ko	22 juin 2021	Jakub Hrozek
1005-python-Skip-the-DSA-key-test-unless-SHA-1-is-configu.patch	1,14 ko	22 juin 2021	Jakub Hrozek
1006-test13_test_lasso_server_load_metadata-Don-t-verify-.patch	1,48 ko	22 juin 2021	Jakub Hrozek