البرنامج النصي لتكوين Azure AD لأمان البريد الإلكتروني من Cisco

المحتويات

المقدمة
المتطلبات الأساسية
المتطلبات
المكونات المستخدمة
البرنامج النصي لتكوين Azure AD لأمان البريد الإلكتروني من Cisco
معلومات ذات صلة

المقدمة

يقدم هذا المستند برنامج نصي يمكن تشغيله من بيئة UNIX/Linux لتبسيط العملية المستخدمة لإنشاء شهادة موقعة ذاتيا وخطوات Microsoft Azure المطلوبة عند الحاجة لتكوين أمان البريد الإلكتروني من Cisco. يمكن إستخدام هذا البرنامج النصي للإصلاح التلقائي لعلبة البريد (MAR) أو موصل LDAP من Microsoft Office 365 أو محلل التهديدات من Cisco ل Office 365. هذا البرنامج النصي مستقل ويمكن إستخدامه مع جميع إصدارات AsyncOS لأجهزة أمان البريد الإلكتروني (ESA).

ملاحظة: تعتبر هذه المادة دليلا على المفهوم، وهي تقدم كمثال على ذلك. وفي حين ان هذه الخطوات جرى إختبارها بنجاح، فإن القصد من هذه المقالة هو بشكل رئيسي الايضاح والإيضاح. البرامج النصية المخصصة خارج نطاق Cisco وقابليتها للدعم. لن يقوم مركز المساعدة التقنية (TAC) ل Cisco بكتابة البرامج النصية الخارجية أو تحديثها أو أستكشاف الأخطاء وإصلاحها في أي وقت. قبل أن تحاول إنشاء أي نصوص تنفيذية، تأكد من أن لديك معرفة البرمجة النصية عندما تقوم ببناء النص النهائي.

ملاحظة: لا يحق ل Cisco TAC و Cisco Support أستكشاف أخطاء جانب العميل وإصلاحها مع Microsoft Exchange أو Microsoft Azure AD أو Office 365.

المتطلبات الأساسية

المتطلبات

توصي Cisco بقراءة وفهم كيفية تكوين إعدادات علبة بريد Azure AD و Office 365 ل ESA.

المكونات المستخدمة

لا يقتصر هذا المستند على إصدارات برامج ومكونات مادية معينة.

لأغراض هذا البرنامج النصي وتنفيذه، من المفترض أن OpenSSL مثبت لديك. من المطالبة الطرفية الخاصة بك، قم بتشغيل **أي إصدار OpenSSL** أو **OpenSSL** للتحقق من التثبيت.

لأغراض هذه المقالة، سيتم إستدعاء النص وتنفيذه على أنه *my_azure.sh*. لا تتردد في تسمية النص كما تشاء.

تم إنشاء المعلومات الواردة في هذا المستند من الأجهزة الموجودة في بيئة معملية خاصة. بدأت جميع الأجهزة المُستخدمة في هذا المستند بتكوين ممسوح (افتراضي). إذا كانت شبكتك قيد التشغيل، فتأكد من فهمك للتأثير المحتمل لأي أمر.

البرنامج النصي لتكوين Azure AD لأمان البريد الإلكتروني من Cisco

من مضيف خارجي (UNIX/Linux)، قم بإنشاء نص تنفيذي وانسخ والصق هذا النص:

```
clear
"########################################################################################" echo
.:|:.:|:. my_azure.sh by Robert Sherwin (robsherw@cisco.com)  ©2018 Cisco
Using openssl, this script will create a self-signed certificate for you to use in
.order to complete the Mailbox Settings configuration for Cisco Email Security
:Please respond to the following prompts
########################################################################################
"
if which openssl >/dev/null; then
echo "openssl check passed: openssl is installed!" & openssl version
else
echo "You do not appear to have openssl installed." && exit
fi

" echo
" :Please enter a name for your cert
read my_cert

;[ while [ -f $my_cert.key
do
echo "File exists, please enter a name for your cert: " && read my_cert
done

" echo
" :Thank you.  The files that will be generated for your cert are

crt=$my_cert.crt
key=$my_cert.key
pem=$my_cert.pem

echo $crt
echo $key
echo $pem
"" echo

while true; do
read -p "Are you ready to proceed and generate these files for your configuration? $(tput
smso)(y/n)$(tput sgr0) " yn
case $yn in
Yy]* ) openssl req -x509 -sha256 -nodes -days 1825 -newkey rsa:2048 -keyout $key -out]
$crt
openssl rsa -in $key -out $key
cat $key $crt > $pem

"" echo
base64Thumbprint=`openssl x509 -outform der -in $crt | openssl dgst -binary -sha1 | openssl
`base64
`base64Value=`openssl x509 -outform der -in $crt | openssl base64 -A
`"(()keyid=`python -c "import uuid; print(uuid.uuid4
" echo
########################################################################
:Next, $(tput smul)copy$(tput rmul) the following to Azure for your manifest
########################################################################
```

```
                                                                                "
                                                                  ] :"\echo "\"keyCredentials
                                                                                  }
                                                          ,"\customKeyIdentifier\": \"$base64Thumbprint"\
                                                                      ,"\keyId\": \"$keyid"\
                                                              ,"\type\": \"AsymmetricX509Cert"\
                                                                      ,"\usage\": \"Verify"\
                                                                "\value\": \"$base64Value"\
                                                                                  {
                                                                                ",[
                                                                             " echo
                    ##############################################################################
      Then $(tput smul)complete$(tput rmul) the Azure configuration to get the $(tput smso)Client
                                        .(ID$(tput sgr0) and $(tput smso)Tenant ID$(tput sgr0
                    ##############################################################################
                                                                                "
                 echo "This is the $(tput smso)Thumbprint$(tput sgr0) for your ESA configuration:
                                                                    "$base64Thumbprint
    echo "This is the $(tput smso)Certificate Private Key$(tput sgr0) for your ESA configuration:
                                                                                $pem
                                                                          ;;break ;"
                                                                   ;;Nn]* ) exit]
                                                        ;;".echo "Please answer yes or no ( *
                                                                             esac
                                                                                done
                                                                  while true; do
read -p "Do you wish to review this certificate in detail? $(tput smso)(y/n)$(tput sgr0) " yn
                                                                  case $yn in
                                                  " Yy]* ) openssl x509 -in $crt -text; echo]
                                                                  ;;Thank you!" && break
                                                    ;;Nn]* ) echo "Thank you!" && exit]
                                                        ;;".echo "Please answer yes or no ( *
                                                                             esac
                                                                                done
```

تلميح: بمجرد كتابة البرنامج النصي، أدخل **chmod u+x <script_name>**لجعل البرنامج النصي قابل للتنفيذ.

يجب أن ينتج عن مثال كامل للنصوص التنفيذية قيد التنفيذ ما يلي:

```
                                                                  my_host$ ./my_azure
##############################################################################################
       .:|:.:|:. my_azure.sh by Robert Sherwin (robsherw@cisco.com)  ©2018 Cisco
   Using openssl, this script will create a self-signed certificate for you to use in
       .order to complete the Mailbox Settings configuration for Cisco Email Security
                                                  :Please respond to the following prompts
##############################################################################################

                                                !openssl check passed: openssl is installed
                                                                  LibreSSL 2.2.7

                                                               :Please enter a name for your cert
                                                                  technote_example

                                    :Thank you.  The files that will be generated for your cert are
                                                                  technote_example.crt
                                                                  technote_example.key
                                                                  technote_example.pem

        Are you ready to proceed and generate these files for your configuration? (y/n) y
                                                          Generating a 2048 bit RSA private key
                                  +++.....................................................
```

```
+++.....................................
'writing new private key to 'technote_example.key
-----
You are about to be asked to enter information that will be incorporated
.into your certificate request
.What you are about to enter is what is called a Distinguished Name or a DN
There are quite a few fields but you can leave some blank
,For some fields there will be a default value
.If you enter '.', the field will be left blank
-----
Country Name (2 letter code) []:US
State or Province Name (full name) []:North Carolina
Locality Name (eg, city) []:RTP
Organization Name (eg, company) []:Cisco
.Organizational Unit Name (eg, section) []:Example Dept
Common Name (eg, fully qualified host name) []:example.local
Email Address []:joe.user@example.local
writing RSA key


################################################################################
:Next, copy the following to Azure for your manifest
################################################################################

] :"keyCredentials"
}
,"=customKeyIdentifier": "wWHhkWEfuhDHTXPzzmHoSEnjbNM"
,"keyId": "338836b8-fc8d-4e1b-9a3f-b252f8368d34"
,"type": "AsymmetricX509Cert"
,"usage": "Verify"
value":"
"MIIDtDCCApwCCQDV3bbiHmaN2jANBgkqhkiG9w0BAQsFADCBmzELMAkGA1UEBhMCVVMxFzAVBgNVBAgMDk5vcnRoIENhcm9
saW5hMQwwCgYDVQQHDANSVFAxDjAMBgNVBAoMBUNpc2NvMRYwFAYDVQQLDA1FeGFtcGxlERlcHQuMRYwFAYDVQQDDA1leGF
tcGxlLmxvY2FsMSUwIwYJKoZIhvcNAQkBFhZqb2UudXNlckBleGFtcGxlLmxvY2FsMB4XDTE4MTAxODAyMDA0OVoXDTIzMTA
xNzAyMDA0OVowgZsxCzAJBgNVBAYTAlVTMRcwFQYDVQQIDA5Ob3J0aCBDYXJvbGluYTEMMAoGA1UEBwwDUlRQMQ4wDAYDVQQ
KDAVDaXNjbzEWMBQGA1UECwwNRXhhbXBsZSBEZXB0LjEWMBQGA1UEAwwNZXhhbXBsZS5sb2NhbDElMCMGCSqGSIb3DQEJARY
Wam9lLnVzZXJAZXhhbXBsZS5sb2NhbDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKlYmW7DN+AxcZQcpc8hZhm
v9yqMHul2cjV3G088mkGtRZU5KUVNKZZSmMlny3lOKg6cTu4Ez4UuigzC/2JXEf3+wOj9YChK92bEYWjYsKeZtbIoqYRfHE+
Sk+bsJb5GpizXgPcYZGje81ecgamhDrg7NZrthPTSKa4ZxmYwpQl6xGDrMipolGoENf+eyNCo5VyAXlxuYH8m6t0GdPw+VKH
J7k+4wI9KTUw4LABoOWs8hUnDi0yz2k9mqNvTG+u75EUUMgcTWc/ISsXjC8kpbOsxteZIiU4xUvqNd1t96iccjad19n61Jds
wGX+CC1Pl+ZZMk8/IQEPtbPqs/4p3cmECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAQqq7ixBbtfhorrWk73uCoYUPRqWZLKH
lgs1UpEnmPjvLZiImY+O6kiR9icDVjFD47AW+0vYg3pHt6pKWl7TUZpilz4hNp0oYc/qjd6aCA8B2KMmbfh2DVhmpYWW8P7w
bNP/im3114F/zJvBVnHjeaY9KsuTUU54Wb8VX2FFX40/YFm/HTHrXcHHyWy5XBU9MFVmEu+Yv6JIxCaEgj5J7jV4qGQM++fn
+EpRPkVHn844Hzgxm40bRW747rjGuyKss+E2tjWJT6OmDJ4ruHCFdvhkZvvzVJyVn0PVN+cwoJ0gLM7p2oa7J3IdNZ3p2CMX
"==vFdZsRiFFUpBIbK3VYlFRrg
{
,[

################################################################################
.Then complete the Azure configuration to get the Client ID and Tenant ID
################################################################################

=This is the Thumbprint for your ESA configuration: wWHhkWEfuhDHTXPzzmHoSEnjbNM
This is the Certificate Private Key for your ESA configuration: technote_example.pem
```

سيطالبك البرنامج النصي بمراجعة الترخيص بالتفصيل. أدخل **y** أو **n** لإكمال البرنامج النصي.

```
Do you wish to review this certificate in detail? (y/n) y
:Certificate
:Data
(Version: 1 (0x0
(Serial Number: 1541067458222060693 8 (0xd5ddb6e21e668dda
Signature Algorithm: sha256WithRSAEncryption
```

```
        Issuer: C=US, ST=North Carolina, L=RTP, O=Cisco, OU=Example Dept.,
                        CN=example.local/emailAddress=joe.user@example.local
                                                              Validity
                                Not Before: Oct 18 02:00:49 2018 GMT
                                Not After : Oct 17 02:00:49 2023 GMT
    Subject: C=US, ST=North Carolina, L=RTP, O=Cisco, OU=Example Dept.,
                        CN=example.local/emailAddress=joe.user@example.local
                                             :Subject Public Key Info
                                 Public Key Algorithm: rsaEncryption
                                          (Public-Key: (2048 bit
                                           :Modulus
                  :a9:58:99:6e:c3:37:e0:31:71:94:1c:a5:cf:21:00
                  :af:f7:2a:8c:1e:e9:76:72:35:77:1b:4f:3c:66:19
                  :9a:41:ad:45:95:39:29:45:4d:29:96:52:98:c9:67
                  :cb:79:4e:2a:0e:9c:4e:ee:04:cf:85:2e:8a:0c:c2
                  :ff:62:57:11:fd:fe:c0:e8:fd:60:28:4a:f7:66:c4
                  :d8:b0:a7:99:b5:b2:28:a9:84:5f:1c:4f:92:61:68
                  :e6:ec:25:be:46:a6:2c:d7:80:f7:18:64:68:de:93
                  :f3:57:9c:81:a9:a1:0e:b8:3b:35:9a:ed:84:f4:d2
                  :ae:19:c6:66:30:a5:09:7a:c4:60:eb:32:2a:68:29
                  :94:6a:04:35:ff:9e:c8:d0:a8:e5:5c:80:5e:5c:6e
                  :60:7f:26:ea:dd:06:74:fc:3e:54:a1:c9:ee:4f:b8
                  :c0:8f:4a:4d:4c:38:2c:00:68:39:6b:3c:85:49:c3
                  :8b:4c:b3:da:4f:66:a8:db:d3:1b:eb:bb:e4:45:14
                  :cf:c8:4a:c5:e3:0b:c9:29:6c:eb:31:32:07:13:59
                  :b5:e6:48:89:4e:31:52:fa:8d:77:5b:7d:ea:27:1c
                  :8d:a7:75:f6:7e:b5:25:db:30:19:7f:82:0b:53:e5
                  :f9:96:4c:93:cf:c8:40:43:ed:6c:fa:ac:ff:8a:77
                                                     72:61
                                       (Exponent: 65537 (0x10001
                                 Signature Algorithm: sha256WithRSAEncryption
                    :aa:bb:8b:10:5b:b5:f8:68:ae:b5:a4:ef:7b:82:a1:85:0f:42
                    :a5:99:2c:a1:e5:82:cd:54:a4:49:e6:3e:3b:cb:66:22:26:46
                    :e3:ba:92:24:7d:89:c0:d5:8c:50:f8:ec:05:be:d2:f6:20:63
                    :de:91:ed:ea:92:96:97:b4:d4:66:98:a5:cf:88:4d:a7:4a:18
                    :fa:a3:77:a6:82:03:c0:76:28:c9:9b:7e:1d:83:56:19:a9:73
                    :bc:3f:bc:1b:34:ff:e2:9b:7d:75:e0:5f:f3:26:f0:55:61:65
                    :9c:78:de:69:8f:4a:b2:e4:d4:53:9e:16:6f:c5:57:d8:51:57
                    :e3:4f:d8:16:6f:c7:4c:7a:d7:70:71:f2:5b:2e:57:05:4f:4c
                    :bb:e6:2f:e8:92:31:09:a1:20:8f:92:7b:8d:5e:2a:15:59:84
                    :19:03:3e:f9:f9:fe:12:94:4f:91:51:e7:f3:8e:07:ce:0c:66
                    :e3:46:d1:5b:be:3b:ae:31:ae:c8:ab:2c:f8:4d:ad:8d:62:53
                    :e8:e9:83:27:8a:ee:1c:21:5d:be:19:19:be:fc:d5:27:25:67
                    :d0:f5:4d:f9:cc:28:27:48:0b:33:ba:76:a1:ae:c9:dc:87:4d
                    :67:7a:76:08:c5:ef:15:d6:6c:46:21:45:52:90:48:6c:ad:d5
                                                        ae:62:51:51
                                                -----BEGIN CERTIFICATE-----
                  MIIDtDCCApwCCQDV3bbiHmaN2jANBgkqhkiG9w0BAQsFADCBmzELMAkGA1UEBhMC
                  VVMxFzAVBgNVBAgMDk5vcnRoIENhcm9saW5hMQwwCgYDVQQHDANSVFAxDjAMBgNV
                  BAoMBUNpc2NvMRYwFAYDVQQLDA1FeGFtcGxlIERlcHQuMRYwFAYDVQQDDA1leGFt
                  cGxlLmxvY2FsMSUwIwYJKoZIhvcNAQkBFhZqb2UudXNlckBleGFtcGxlLmxvY2Fs
                  MB4XDTE4MTAxODAyMDA0OVoXDTIzMTAxNzAyMDA0OVowgZsxCzAJBgNVBAYTAlVT
                  MRcwFQYDVQQIDA5Ob3J0aCBDYXJvbGluYTEMMAoGA1UEBwwDUlRQMQ4wDAYDVQQK
                  DAVDaXNjbzEWMBQGA1UECwwNRXhhbXBsZSBEZXB0LjEWMBQGA1UEAwwNZXhhbXBs
                  ZS5sb2NhbDElMCMGCSqGSIb3DQEJARYWam9lLnVzZXJAZXhhbXBsZS5sb2NhbDCC
                  ASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAKlYmW7DN+AxcZQcpc8hZhmv
                  9yqMHul2cjV3G088mkGtRZU5KUVNKZZSmMlny3lOKg6cTu4Ez4UuigzC/2JXEf3+
                  wOj9YChK92bEYWjYsKeZtbIoqYRfHE+Sk+bsJb5GpizXgPcYZGje81ecgamhDrg7
                  NZrthPTSKa4ZxmYwpQl6xGDrMipolGoENf+eyNCo5VyAXlxuYH8m6t0GdPw+VKHJ
                  7k+4wI9KTUw4LABoOWs8hUnDi0yz2k9mqNvTG+u75EUUMgcTWc/ISsXjC8kpbOsx
                  teZIiU4xUvqNd1t96iccjad19n61JdswGX+CC1Pl+ZZMk8/IQEPtbPqs/4p3cmEC
                  AwEAATANBgkqhkiG9w0BAQsFAAOCAQEAQqq7ixBbtfhorrWk73uCoYUPRqWZLKHl
                  gs1UpEnmPjvLZiImY+O6kiR9icDVjFD47AW+0vYg3pHt6pKWl7TUZpilz4hNp0oY
                  c/qjd6aCA8B2KMmbfh2DVhmpYWW8P7wbNP/im3114F/zJvBVnHjeaY9KsuTUU54W
```

```
+b8VX2FFX40/YFm/HTHrXcHHyWy5XBU9MFVmEu+Yv6JIxCaEgj5J7jV4qGQM++fn
EpRPkVHn844Hzgxm40bRW747rjGuyKss+E2tjWJT6OmDJ4ruHCFdvhkZvvzVJyVn
==0PVN+cwoJ0gLM7p2oa7J3IdNZ3p2CMXvFdZsRiFFUpBIbK3VYlFRrg
-----END CERTIFICATE-----

!Thank you
```

في هذا الوقت، لديك ثلاثة ملفات: .key، .crt، و .pem.

أستخدم إخراج *keyCredentials* كما تم توجيهه، وانسخ الإخراج إلى Azure عند إعداد تسجيل التطبيق. يلزم إخراج *بصمة الإبهام ومفتاح الشهادة الخاص* (.pem) عند تشغيل خطوات التكوين على أمان البريد الإلكتروني من Cisco.

معلومات ذات صلة

- جهاز أمان البريد الإلكتروني من Cisco - أدلة المستخدم النهائي
- الدعم التقني والمستندات - Cisco Systems

حول هذه الترجمة
ترجمت Cisco هذا المستند باستخدام مجموعة من التقنيات الآلية والبشرية لتقديم محتوى دعم للمستخدمين في جميع أنحاء العالم بلغتهم الخاصة. يُرجى ملاحظة أن أفضل ترجمة آلية لن تكون دقيقة كما هو الحال مع الترجمة الاحترافية التي يقدمها مترجم محترف. تخلي Cisco Systems مسئوليتها عن دقة هذه الترجمات وتُوصي بالرجوع دائمًا إلى المستند الإنجليزي الأصلي (الرابط متوفر).