

Postfix と Kannel に基づく SMS ゲートウェイとの ISE ゲスト パスワード統合の設定例

目次

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[設定](#)

[SMS ゲートウェイのメリット](#)

[ネットワーク ダイアグラムとトラフィックフロー](#)

[設定](#)

[ISE](#)

[Postfix](#)

[Mailfilter を使用した Maildrop](#)

[Kannel](#)

[確認](#)

[ISE](#)

[Postfix](#)

[Maildrop](#)

[Mailfilter](#)

[Kannel](#)

[ゲストの電話機](#)

[トラブルシューティング](#)

[ISE](#)

[関連情報](#)

概要

このドキュメントでは、ゲスト アカウントを使用してショート メッセージ サービス (SMS) メッセージをユーザに配信するために、オープン ソース ソリューション (Postfix、Maildrop、Kannel) と Cisco Identity Services Engine (ISE) を統合する方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Cisco ISE とゲスト アクセス
- Linux とシェルスクリプティング

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- Cisco ISE バージョン 1.2 以降
- Postfix バージョン 2.10
- Maildrop バージョン 2.6.0
- Kannel バージョン 1.5.0

注: Postfix、Maildrop、および Kannel はオープン ソース ソリューションであり、シスコはこれらの製品をサポートしていないことに注意してください。この設定例では、ISE と他の製品を統合してエンドツーエンド ソリューションを提供する方法を簡単に説明します。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、クリアな (デフォルト) 設定で作業を開始しています。ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

背景説明

ISE を使用すれば、主に、ゲスト、ビジター、請負業者、コンサルタント、およびカスタマーの一時的なネットワーク アクセス用のゲスト アカウントを作成することができます。このようなアカウントは、スポンサー ポータル経由でスポンサー ユーザが作成します。アカウントを作成したら、動的に生成されたアクセス パスワードを SMS 経由で直接ゲスト ユーザの携帯電話に送信することができます。

Cisco ISE は、Simple Mail Transfer Protocol (SMTP) を使用した電子メール経由でこれらのクレデンシャルを Mail2SMS ゲートウェイに送信することができます。このゲートウェイが SMS の配信を担当します。

設定

SMS ゲートウェイのメリット

さまざまな Mail2SMS ゲートウェイ ソリューションが市販されています。これらの多くは、SMTP、Short Message Peer-to-Peer (SMPP)、FTP、HTTP (Simple Object Access Protocol (SOAP)、Web サービス) などのさまざまなプロトコルを使用してデータを受信し、SMS メッセージを特定の携帯電話に送信できます。

独自の SMS ゲートウェイを構築することをお勧めします。次のことが可能になります。

- 柔軟性の向上
- ルーティング (タイム ベース、ポリシー ベース、コンテンツ ベース) に関する複合ルールを構築可能な機能
- ローカル データベース (Active Directory グループごとに異なるルーティング ポリシーなど) との統合
- 運用コストの削減 (外部サービスに関するコストが不要)
- ISE によって生成され、電子メールとして送信されるヘルス アラートにもこのソリューションを使用できる可能性

混合展開 (外部サービスにも統合された個人用の SMS ゲートウェイ) を使用する価値があります。

ネットワーク ダイアグラムとトラフィック フロー

ここで、フローを示します。

1. スポンサー ユーザが SMS 通知を使ってゲスト アカウントを作成し、ユーザの携帯電話番号を指定します。ISE が電子メールを設定された SMTP サーバに送信します。送信元アドレス (**From**) は特定のスポンサー ユーザに属しているのに対して、宛先アドレス (**To**) は ISE 上でグローバルに設定されます (この例では、`sms@test-cisco.com`)。新しく作成されたユーザに関する詳細 (ユーザ名やパスワードなど) のすべてが電子メールの本文に含まれます。
2. 電子メールが Postfix サーバに到着すると、maildrop でローカル配信エージェントとして設定されます。SMS ユーザの maildir ディレクトリに配信される直前に、maildrop がユーザのホーム ディレクトリで mailfilter を検索します。mailfilter スクリプトが電子メールを解析して、必要なすべてのデータが見つかったら、`wget` を使用して、HTTP GET 要求を Kannel smsbox に送信します。この HTTP GET 要求には、ユーザ名とパスワードを伴うテキストメッセージと、ユーザの携帯電話番号が含まれています。Kannel smsbox は、ユーザからのすべての要求を受け入れて SMS を送信する (Kannel bearerbox に渡す) ために使用される Kannel のフロントエンドです。
3. Kannel smsbox が SMS の送信を担当する Kannel bearerbox にその要求を送信します。
4. bearerbox 上で、複数のルールとショート メッセージ サービス センター (SMSC) が設定されている場合があります。次の例では、外部 SMPP サーバを使用します。ローカル接続された携帯電話の設定は簡単であり、後述します。

このソリューションのモジュール (Postfix、Kannel smsbox、および Kannel bearerbox) ごとに別々のサーバにインストールできます。この例では、分かりやすくするために、同じサーバ上で設定します。

設定

ISE

ISE を設定するには、次の手順を実行します。

1. スポンサー ポータル ユーザを設定します。この例では、デフォルト ISE 設定が使用され、ユーザは **SponsorAllAccount** グループに配属されます。

スポンサー ユーザの電子メールは、後で、スポンサー ポータルから設定できます。

2. SMS 通知を送信できるようにするために、**SponsorAllAccount** グループのデフォルト権限を編集します。

デフォルトで、**Send SMS** 権限は無効になっています。

3. SMTP サーバを設定して、DNS 設定が正しいことを確認します。

すべての通知電子メールが **smtp.test-cisco.com** ホストに送信されます。ISE は、設定されたドメインの DNS MX レコードをチェックしません (この SMTP サーバはリレーとして扱われます)。

4. SMS 通知として送信される電子メールをカスタマイズします。

5. 宛先電子メール アドレスを設定します。これがデフォルトから変更する唯一の設定です。通知はすべて、**To** フィールドが **sms@test-cisco.com** に設定され、以前設定した SMTP サーバ経由で送信されます。

注: 電子メール経由でアラート通知を送信するように ISE を設定することができます。これは、提案するソリューションと統合して、アラートを SMS として携帯電話に送信することもできます。Postfix サーバでは、これ専用のアカウント (**alert@test-cisco.com** など) を使用することをお勧めします。

Postfix

Postfix は、ISE から電子メールを受信する SMTP サーバです。いくつかのわずかな変更を除いて、デフォルト設定が使用されます。これを設定するには、次の手順を実行します。

1. **test-cisco.com** ドメインのローカル宛先として Postfix を設定します。ローカル配信エージェントの **maildrop** を設定することも重要です。ここで、**main.cf** 内の必要な変更を示します。

```
myhostname = smtp.test-cisco.com
mydomain = test-cisco.com
mydestination = $myhostname, $mydomain, localhost
local_transport = maildrop
```

2. 次のステップは、master.cf 内で maildrop をアクティブにすることです。master.cf 内の該当する行を変更します。

```
maildrop unix - n n - - pipe
  flags=DRhu user=vmail argv=/usr/bin/maildrop -d ${user}
```

これは仮想ドメインを使用しないシンプルな展開のため、デフォルトの {recipient} パラメータの代わりに、{user} パラメータが使用されます。

3. 電子メールの受信に使用されるローカル アカウント SMS を設定します。

```
neptun ~ # useradd sms
neptun ~ # passwd sms
New password:
BAD PASSWORD: it is too simplistic/systematic
Retype new password:
passwd: password updated successfully
neptun ~ # chown -R sms:sms /home/sms/
```

これで、すべての電子メールが SMS ユーザに正しく配信されるはずです。初めて電子メールを受信したときに maildir 構造が自動的に作成されます。

Mailfilter を使用した Maildrop

配信の直前に、maildrop がユーザのホーム ディレクトリで .mailfilter を検索します。そのファイルが見つかり、スクリプトが実行されます。ファイルに関する権限はユーザのみに限定する必要があります。

```
neptun sms # touch /home/sms/.mailfilter
neptun sms # chmod go-rwx /home/sms/.mailfilter
ここで、ファイルの内容を示します。
```

```
# Mailfilter script for parsing ISE SMS messages
# Author: Michal Garcarz at cisco.com
# Date: 1 Dec 2013

#DEFAULT="$HOME/.maildir/"
DATE=`date`
SHELL="/bin/bash"

# Our log file
logfile "/home/sms/maildrop.log"

# Our verbosity in the log file
VERBOSE="5"

log "-----SMS MAILFILTER LOG-----"
log "Email received at: $DATE"

if (/^Subject:. *Guest.*Text.*Notification.*/)
{
    log "Email processed by script sending SMS via Kannel"

    USERNAME=""
    PASSWORD=""
    TO=""
```

```

if (/^text:Username:(.*)/:b)
{
    log "Username exists $MATCH1"
    USERNAME=$MATCH1
}
if (/^text:Password:(.*)/:b)
{
    log "Password exists $MATCH1"
    PASSWORD=$MATCH1
}
if (/^to:(.*)/:b)
{
    log "Mobile phone exists $MATCH1"
    TO=$MATCH1
}

if ($USERNAME ne "" && $PASSWORD ne "" && $TO ne "")
{
    log "Sending via HTTP to kannel username=$USERNAME password=$PASSWORD to=$TO"
    DATA="ISE Guest portal Username: $USERNAME Password: $PASSWORD"

    #also curl can be used instead of wget
    xfilter "wget -O/dev/null \"http://192.168.112.100:13013/cgi-bin/sendsms?username=
tester&password=foobar&to=$TO&text=$DATA\" >> /tmp/mailedrop-kannel.log 2>>
/tmp/mailedrop-kannel.log"
}

#deliver to maildir (not used since xfilter returns !=0)
to $DEFAULT/
}

```

スクリプトの動作は次のとおりです。

- 件名が ISE 上で設定されたものと同じかどうかをチェックする。
- ユーザ名とパスワードを電子メール本文のフィールドに読み込む (ISE からのデフォルト テンプレートが使用される)。
- すべてのフィールドが存在する場合は、外部プログラムの **wget** を呼び出して、すべてのパラメータと一緒に HTTP GET を Kannel smsbox に送信する。 URL に特定のクレデンシャル (username=tester&password=foobar) が使用されていることに注意してください。これらが、SMS を送信する権限を持っている、Kannel 内で設定されたユーザのクレデンシャルです。

次の 2 つのログ ファイルがあります。

- //home/sms/mailedrop.log : スクリプトの実行からのログ
- //tmp/mailedrop-kannel.log : wget の実行からのログ

Kannel

smsbox と bearerbox の両方を 1 つのファイルで設定できます。この設定では、配信用の外部 SMPP サーバが使用されます。「**smpp sms service provider**」というフレーズを検索すると、Web 上で複数のサービスが簡単に見つかります。SMS メッセージを受信してルーティングする必要がないため、設定はシンプルです。このソリューションは送信専用であり、1 つの SMPP プロバイダーを使用します。

ここで、`/etc/kannel/kannel.conf` からの抜粋を示します。

```
#bearerbox
group = core
admin-port = 13000
admin-password = bar
smsbox-port = 13001
log-level = 0
log-file = "/var/log/kannel/kannel.log"
access-log = "/var/log/kannel/access.log"
```

```
# SMSC SMPP
group = smsc
smc = smpp
host = ****.com
port = 1775
smc-username = 41jt7wi3
smc-password = *****
system-type =
address-range =
```

```
# SMSBOX SETUP
group = smsbox
bearerbox-host = localhost
sendsms-port = 13013
sendsms-chars = "0123456789 +-"
global-sender = 12345
log-file = "/var/log/kannel/smsbox.log"
log-level = 0
access-log = "/var/log/kannel/access.log"
white-list =
black-list =
reply-couldnotfetch =
reply-couldnotrepresent =
reply-requestfailed =
reply-emptymessage =
```

```
# SEND-SMS USERS, this credentials has been used in wget script
group = sendsms-user
username = tester
password = foobar
user-deny-ip = "*. *.*.*"
user-allow-ip = "192.168.*.*"
```

```
# SMS SERVICE Default
# there should be default always (this is for receiving SMS messages - not used)
group = sms-service
keyword = default
text = "No service specified"
```

携帯電話を USB 経由で接続して、GSM SMSC を設定することができます。

```
group = smsc
smc = at #type = GSM
smc-id = usb0-modem
my-number = 1234
modemtype = auto #types: wavcom, siemens, siemens-tc35, falcom,
nokiaphone, ericsson
device = /dev/ttyUSB0 #phone device seen on server
```

ほとんどの電話機で、モデム機能をアクティブにする必要もあります。たとえば、Android バージョン 2.2 以降では、[Settings] > [Tethering] と [Portable Hotspot] > [USB tethering] で有効にします。

bearerbox と smsbox の両方を実行することを忘れないでください。次に例を示します。

```
neptun ~ # /etc/init.d/kannel-bearerbox start
* Starting kannel bearerbox ... [ ok ]
neptun ~ # /etc/init.d/kannel-smsbox start
* Starting kannel smsbox ... [ ok ]
```

```
neptun ~ # netstat -atcpn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address      Foreign Address    State    PID/Program name
tcp      0      0 0.0.0.0:13013      0.0.0.0:*           LISTEN   24170/smsbox
tcp      0      0 0.0.0.0:13000      0.0.0.0:*           LISTEN   24151/bearerbox
tcp      0      0 0.0.0.0:13001      0.0.0.0:*           LISTEN   24151/bearerbox
```

bearerbox を開始するためには、1 つ以上の SMSC が設定されている必要があります。

確認

このセクションでは、設定が正常に機能していることを確認します。

ISE

スポンサー ポータルのデフォルト ポート (8443) が使用されます。スポンサーが <https://ise.test-cisco.com:8443/sponsorportal/> にログインします。

[My Settings] で、スポンサーに電子メール アドレスが割り当てられていることを確認します。

SMS 通知を使ってゲスト アカウントを作成します。

ゲスト アカウントが正常に作成されたことを伝える確認通知が送られてきます。

ISE が電子メールを設定された SMTP サーバに送信するはずです。

Postfix

SMTP サーバがメッセージを受信して、maildrop 経由でローカル アカウント (sms@test-cisco.com) に配信します。ここで、`/var/log/messages` からの抜粋を示します。

```
Nov 30 22:39:47 neptun postfix/smtpd[18460]: connect from unknown[192.168.112.1]
Nov 30 22:39:47 neptun postfix/smtpd[18460]: 2B36030B32: client=unknown
[192.168.112.1]
Nov 30 22:39:47 neptun postfix/cleanup[18463]: 2B36030B32: message-id=
<563762958.941385847586377.JavaMail.root@ise2>
Nov 30 22:39:47 neptun postfix/qmgr[32658]: 2B36030B32: from=<sponsor@test-cisco.com>,
size=689, nrcpt=1 (queue active)
Nov 30 22:39:47 neptun postfix/pipe[18464]: 2B36030B32: to=<sms@test-cisco.com>,
relay=maildrop, delay=0.18, delays=0.14/0/0/0.04, dsn=2.0.0, status=sent (delivered
via maildrop service)
```

Maildrop

電子メールを SMS に送信する前に、maildir ディレクトリが `/home/sms/mailfilter` を実行して、特定のアクションが起動されます。

ここで、/home/sms/mailedrop.log からの抜粋を示します。

```
-----SMS MAILFILTER LOG-----  
Email received at: Sat Nov 30 22:39:47 CET 2013  
Email processed by script sending SMS via Kannel  
Username exists jsmith02  
Password exists t6ub79_6r  
Mobile phone exists 4850xxxxxxx  
Sending via HTTP to kannel username= jsmith02 password= t6ub79_6r to=4850xxxxxxx
```

Mailfilter

mailfilter スクリプトはすべてのデータを読み取って xfilter を実行します。その結果、wget が呼び出され、すべてのパラメータが Kannel に渡されます。

ここで、/tmp/mailedrop-kannel.log からの抜粋を示します。

```
--2013-11-30 22:39:47-- http://192.168.112.100:13013/cgi-bin/sendsms?username=  
tester&password=foobar&to=4850xxxxxxx&text=ISE%20Guest%20portal%20Username:  
%20%20jsmith02%20Password:%20%20t6ub79_6r  
Connecting to 192.168.112.100:13013... connected.  
HTTP request sent, awaiting response... 202 Accepted  
Length: 24 [text/html]  
Saving to: `/dev/null'
```

OK

100% 1.14M=0s

2013-11-30 22:39:47 (1.14 MB/s) - `/dev/null' saved [24/24]

HTTP GET 要求が受け付けられています。

Kannel

Kannel smsbox は、wget から HTTP 要求を受信して、SMS を配信するためにその要求を bearerbox に送信したことを報告します。

ここで、/var/log/kannel/smsbox.log からの抜粋を示します。

```
2013-11-30 22:39:47 [18184] [5] INFO: smsbox: Got HTTP request </cgi-bin/sendsms>  
from <192.168.112.100>  
2013-11-30 22:39:47 [18184] [5] INFO: sendsms used by <tester>  
2013-11-30 22:39:47 [18184] [5] INFO: sendsms sender:<tester:12345>  
(192.168.112.100) to:<4850xxxxxxx> msg:<ISE Guest portal Username:  
jsmith02 Password: t6ub79_6r>  
2013-11-30 22:39:47 [18184] [5] DEBUG: Stored UUID fd508632-9408-49e1-9eda-  
3ce8d4b939d4  
2013-11-30 22:39:47 [18184] [5] DEBUG: message length 57, sending 1 messages  
2013-11-30 22:39:47 [18184] [5] DEBUG: Status: 202 Answer: <Sent.>  
2013-11-30 22:39:47 [18184] [5] DEBUG: Delayed reply - wait for bearerbox  
2013-11-30 22:39:47 [18184] [0] DEBUG: Got ACK (0) of fd508632-9408-49e1-9eda-  
3ce8d4b939d4
```

Kannel bearerbox がリモート SMPP サーバに接続して、メッセージが正常に送信されています。

ここで、/var/log/kannel/kannel.log からの抜粋を示します。

```
2013-11-30 22:39:47 [18165] [8] DEBUG: boxc_receiver: sms received  
2013-11-30 22:39:47 [18165] [8] DEBUG: send_msg: sending msg to box: <127.0.0.1>
```

```

2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
throughput (0.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
Sending PDU:
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU 0x2056bf0 dump:
2013-11-30 22:39:47 [18165] [6] DEBUG:   type_name: submit_sm
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_id: 4 = 0x00000004
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_status: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sequence_number: 5 = 0x00000005
2013-11-30 22:39:47 [18165] [6] DEBUG:   service_type: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr_ton: 2 = 0x00000002
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr_npi: 1 = 0x00000001
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr: "12345"
2013-11-30 22:39:47 [18165] [6] DEBUG:   dest_addr_ton: 2 = 0x00000002
2013-11-30 22:39:47 [18165] [6] DEBUG:   dest_addr_npi: 1 = 0x00000001
2013-11-30 22:39:47 [18165] [6] DEBUG:   destination_addr: "4850xxxxxxxx"
2013-11-30 22:39:47 [18165] [6] DEBUG:   esm_class: 3 = 0x00000003
2013-11-30 22:39:47 [18165] [6] DEBUG:   protocol_id: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   priority_flag: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   schedule_delivery_time: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   validity_period: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   registered_delivery: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   replace_if_present_flag: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   data_coding: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sm_default_msg_id: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sm_length: 57 = 0x00000039
2013-11-30 22:39:47 [18165] [6] DEBUG:   short_message:
2013-11-30 22:39:47 [18165] [6] DEBUG:     Octet string at 0x205ec70:
2013-11-30 22:39:47 [18165] [6] DEBUG:       len: 57
2013-11-30 22:39:47 [18165] [6] DEBUG:       size: 58
2013-11-30 22:39:47 [18165] [6] DEBUG:       immutable: 0
2013-11-30 22:39:47 [18165] [6] DEBUG:       data&colon; 49 53 45 20 47 75 65 73
74 20 70 6f 72 74 61 6c   ISE Guest portal
2013-11-30 22:39:47 [18165] [6] DEBUG:       data&colon; 20 55 73 65 72 6e 61 6d
65 3a 20 20 6a 73 6d 69   Username: jsmi
2013-11-30 22:39:47 [18165] [6] DEBUG:       data&colon; 74 68 30 32 20 50 61 73
73 77 6f 72 64 3a 20 20   th02 Password:
2013-11-30 22:39:47 [18165] [6] DEBUG:       data&colon; 74 36 75 62 37 39 11 36
72   t6ub79.6r
2013-11-30 22:39:47 [18165] [6] DEBUG:     Octet string dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
throughput (1.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
throughput (1.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
Got PDU:
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU 0x2056bf0 dump:
2013-11-30 22:39:47 [18165] [6] DEBUG:   type_name: submit_sm_resp
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_id: 2147483652 = 0x80000004
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_status: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sequence_number: 5 = 0x00000005
2013-11-30 22:39:47 [18165] [6] DEBUG:   message_id: "4128473611307259"
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3]:
throughput (1.00,0.00)

```

送信元アドレスが **12345** に設定されていることに注意してください。この設定は問題ではありません。外部 SMPP サーバがこの値を書き換えます。別に提供される付加サービスを購入することができます。

ゲストの電話機

ゲスト ユーザが SMS を受信します。

トラブルシューティング

このセクションでは、設定のトラブルシューティングに役立つ情報を提供します。

ISE

ゲスト アカウントを作成しているときに、次のエラーが発生する場合があります。 Unable to send a text message to the following guest users: xxxx. 設定ページに電子メール アドレスを追加する必要があります。 このエラー メッセージが表示された場合は、スポンサーの電子メール アドレスを確認します。

関連情報

- [Cisco Identity Services Engine - ユーザ ガイド 1.2](#)
- [Kannel のマニュアル](#)
- [Postfix のマニュアル](#)
- [テクニカルサポートとドキュメント - Cisco Systems](#)