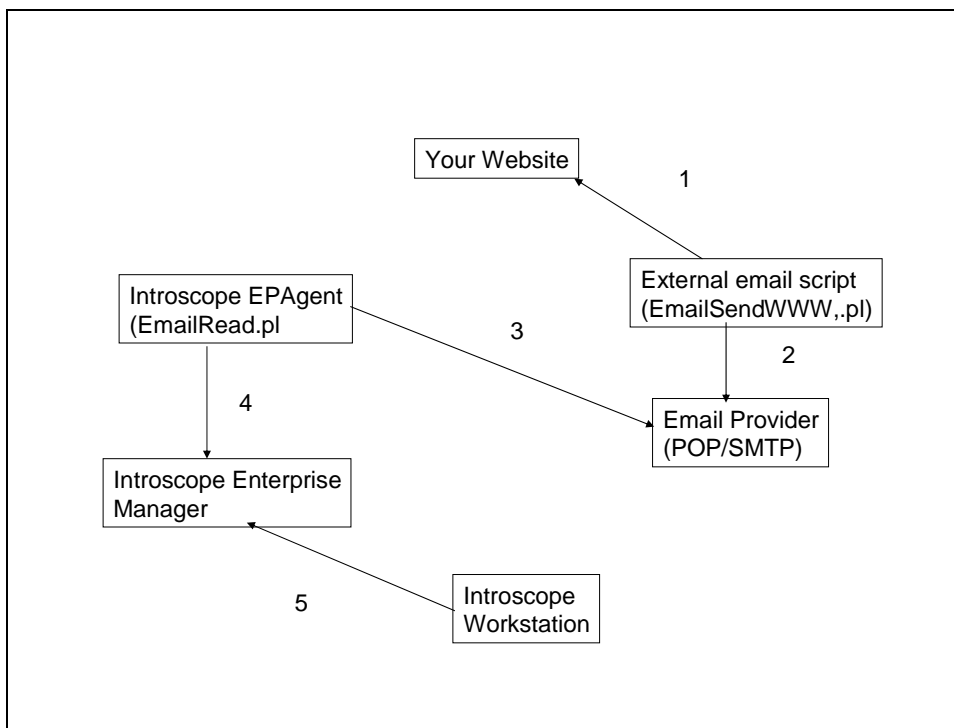## External Email Agent

Sometimes the only true test of a websites availability and end user performance is to test downloading a page from outside the company network.

A stand alone perl script was created to download a page and email key metrics back to an email address. Another perl script runs on an EPAgent which checks the email account downloads the metrics from the email and sends them into the Introscope Enterprise Manager.

This provides an outside network view of the some key SLA metrics.

Design

Here is an overview of the logical flow.

1. The external script (EmailSendWWW.pl) performs a wget on the requested website.
2. The results from the http request are sent via the SMTP to an email account.
3. The EPAgent running within the companies network, accessed the email account using pop3 on a schedule determined by the delay seconds parameter.
4. The email is read and verified from a valid email address and the format for the message is fixed to make spoofing more difficult.
5. The valid metrics are sent into the Enterprise Manager, where they can be seen using the Workstation.

Configuration

The script has a number of variables that must be edited for the environment. These can be found in the source.

Variable Name	Default Values	Description
$RESPdata	/tmp/EmailHttp	Temporary file for returned data. Overwritten on each invocation.
$RESPout	tmp/Emailout	Temporary file for standard out. Overwritten on each invocation.
$url	http://YOUR-SITE	Target url to test.
$from	YOU@YOURS.COM	Email from address.
$site	YOUR-DOMAIN	Site name where email is sent from.
$smtp_host	SMTP-HOSTNAME	Email server name.
$to	TARGET-EMAIL-ADDRESS	Email recipient.
$validuser		Metrics from this user ONLY will be processed.

Security Configuration

Consider the security implications of this Email server information and protect the file so that it can only be read and executed by an authorized user.

Consider creating a standalone email account (not shared) for Introscope to use.

Only metrics from the username is $validuser with the subject "Introscope External Agent Report" will be processed, this makes it much more difficult for incorrect metrics to be processed by the EmailRead.pl script.

The EmailRead.pl script does require a valid user account and password to read email; this should be secured so that it cannot be misused.

Returned Metrics

The following Metrics are returned to Introscope Enterprise Manager.

Type	Name	Value	Description
StringEvent	External:LastCheck	$RESPtime	Timestamp of last check
LongCounter	External:SpeedKBs	$RESPspeed	Speed of response in KB/s for page requested
LongCounter	External:PageSize	$RESPsize	Number of bytes downloaded

Running the EPAgent

This requires the EmailSendWWW.pl to execute outside the company network using perl and the EmailRead.pl to run as a standard EPAgent command.

To execute the script outside of the company network, run the following command.

```
perl EmailSendWWW.pl
```

The EmailRead.pl must be defined to the Introscope EPAgent properties file, an example is shown below.

```
introscope.epagent.plugins.stateless.names=EMAILREAD
introscope.epagent.stateless.EMAILREAD.command=perl EmailRead.pl
introscope.epagent.stateless.EMAILREAD.delayInSeconds=900
```

Design Assumptions

Only email messages from $validuser with the subject "*Introscope External Agent Report*" will be processed, once processed the message will be deleted.

The format of the email header is checked at specific locations, this means that if any other mail is sent it is ignored. Only email sent in the format which EmailSendWWW.pl uses is considered valid.

Metrics are processed in the order they are retrieved from the POP3 mail provider, so ideally the execution and checking time should not overlap.

Prerequisites

This script requires the following items to be available within the environment.

1. Perl
2. Wily Introscope EPAgent
3. Net::POP3 perl module available from www.cpan.org

Perl Script

The following is the provided script example, this is delivered as-is with no warranty.

EmailRead.pl

```perl
#!/bin/perl

use Net::POP3;

$validuser = "ONLY-THIS-USER-IS-OK";
$popserver = 'YOUR-POP-SERVER-NAME';
$popusername = 'YOUR-POP-USER-ACCOUNT-NAME';
$popuserpassword = 'YOUR-POP-USER-ACCOUNT-PASSWORD';

    $pop = Net::POP3->new($popserver, Timeout => 60, Debug=>0);

    if ($pop->login($popusername, $popuserpassword) > 0) {
      my $msgnums = $pop->list;

      foreach my $msgnum (keys %$msgnums) {
        my $msg = $pop->top($msgnum,5);

      if (@$msg[11] =~ /\bIntroscope External Agent Report\b/s or
@$msg[8] =~ /\b$validuser\b/s) {

            my $msgbody = $pop->top($msgnum,5);

            if (@$msgbody[15] =~ /\bmetric\b/s) {

                for ($i=15; $i<20; $i++) {
                        if (@$msgbody[$i] ne "" ) {
                        chomp(@$msgbody[$i]);
                        print "@$msgbody[$i]\n"; }
                }
                $pop->delete($msgnum);
            }
      }
      }
    }
    $pop->quit;
```

EmailSendWWW.pl

```perl
#!/bin/perl

use Net::SMTP;

my $RESPdata = "/tmp/EmailHttp";
my $RESPout  = "/tmp/Emailout";
my $url = "TARGET-URL";
my $from = 'YOUR-VALID-USER-NAME';
my $site = 'YOUR-SITE-DOMAIN-NAME';
my $smtp_host = 'NAME-OF-SMTP-SERVER';
my $to = 'EMAIL-ADDRESS-TO-SEND-RESULTS-TOO';
```

```perl
`wget -o $RESPout -O $RESPdata $url`;
open RESP, $RESP;

foreach (<RESP>)
        {
        if (/\bsaved\b/s)
        {
                $RESPtime = substr($_,0, 8);
                $RESPspeed = substr($_, (index($_, "(")+1),(rindex($_, "
KB") -( index($_, "(")+1)-3));
                $RESPsize = substr($_, (index($_, "[")+1), (rindex($_, "]")
-( index($_, "[")+1)));

        }
}

close RESP;

my $smtp = Net::SMTP->new($smtp_host, Hello => $site, Debug=>0);

$smtp->mail($from);
$smtp->to($to);
$smtp->data();

$smtp->datasend("To: $to\n");
$smtp->datasend("Subject: Introscope External Agent Report\n");
$smtp->datasend("\n");
$smtp->datasend("<metric type=\"StringEvent\"
name=\"External:LastCheck\" value=\"$RESPtime\"/>\n");
$smtp->datasend("<metric type=\"LongCounter\"
name=\"External:SpeedKBs\" value=\"$RESPspeed\"/>\n");
$smtp->datasend("<metric type=\"LongCounter\"
name=\"External:PageSize\" value=\"$RESPsize\"/>\n");
$smtp->dataend();
$smtp->quit;
```