Network Working Group   R. Gellens
Request for Comments: 4146   QUALCOMM
Category: Informational   August 2005

                     Simple New Mail Notification

Abstract

   This memo documents a long-standing technique, supported by a large
   number of mail servers, which allows users to be notified of new
   mail. In addition to server support, there are a number of clients
   that support this, ranging from full email clients to specialized
   clients whose only purpose is to receive new mail notifications and
   alert a mail client.

   In brief, the server sends the string "nm_notifyuser" CRLF to the
   finger port on the IP address (either configured or last used) of the
   user who has received new mail.

Table of Contents

1. Introduction

   There is a long-standing technique supported by a large number of
   mail servers that allows users to be notified of new mail. In
   addition to server support, there are a number of clients that
   support this, ranging from full email clients to specialized clients
   whose only purpose is to receive new mail notifications and alert a
   mail client. This technique is sometimes known as "notify mail"
   (after a shareware client of the same name), "biff", and the "finger
   hack".

2. Conventions Used in This Document

   In examples, "C:" indicates lines sent by the client, and "S:"
   indicates those sent by the server. Line breaks within a command
   example are for editorial purposes only. Line breaks in the protocol
   are indicated by "CRLF".

3. Simple Mail Notification

   With this technique, the server sends the string "nm_notifyuser"
   immediately followed by CRLF to the finger port on the IP address for
   the user who has received new mail. The finger port is 79. Note
   that only the port for finger is used; the finger protocol itself is
   not used.

   The IP address to use may be configured, or the server may use the IP
   address that was last used to check mail by the user. Typically,
   this is a per-account configuration option.

   On the client system, a process must be listening to the finger port
   to be useful. When it receives the "nm_notifyuser" string, it takes
   a configured action, typically instructing a mail client to fetch
   mail.

   Normally, a TCP connection to the target computer is opened, the
   "nm_notifyuser" string is sent, and the connection is closed without
   waiting for a response.

   In some cases, UDP is used instead of TCP, but the default and
   general practice is TCP. Even though only a single message in one
   direction is sent (with no reply), TCP is used most often, probably
   for reliability.

   There is an assumption that the client listening on an IP address
   only has responsibility for one email account. If a client can check
   multiple accounts and receives the "nm_notifyuser" string, it does
   not know which account received the mail.

There is a requirement that a finger daemon not be active on the
client.

4. Example

This example assumes that new mail has arrived at the server
mail.isp.example.com for account fastness@example.net. The server
has determined an IP address to which notification should be sent.

```
C: <listens on TCP port 79>
S: <opens TCP connection to IP address port 79>
C: <accepts inbound connection on port 79>
S: nm_notifyuserCRLF
S: <closes TCP connection>
```

5. Security Considerations

There is no assurance that the "nm_notifyuser" message is being sent
to the correct IP address. Nor does the listening agent on the
client system have any assurance that an "nm_notifyuser" string was
sent by a mail server that has received new mail for the user.

It is trivial for an attacker to send large numbers of
"nm_notifyuser" messages to a targeted system. Client systems that
are listening for this message SHOULD implement protections against
being flooded with notifications. Many server systems already
implement protections against users logging in and checking mail too
frequently.

Because use of this protocol requires that a port be open with an
agent listening on it, if that agent contains vulnerabilities, it may
create a remotely exploitable attack (for example, buffer overflows
that permit an attacker to execute arbitrary code on the client
system with the permissions of the agent). As usual, with a process
listening on a port, the process should execute with the least
possible privilege level and access.

6. IANA Considerations

The notify mail hack (and this document) should be included as an
additional usage for port 79.

7. Acknowledgments

The NotifyMail shareware utility was written by Scott Gruby.

Author's Address

     Randall Gellens
     QUALCOMM Incorporated
     6455 Lusk Blvd.
     San Diego, CA 92121-2779
     USA
     EMail: randy@qualcomm.com

Full Copyright Statement

Intellectual Property

Acknowledgement