



# Supporting Unsupported Protocols: SMTP, POP3 and VoIP

---

First Presented for:

*Rational User's Conference*

*Orlando, FL 2002*

with:

*Chris Walters*

Scott Barber

Chief Technology Officer

PerfTestPlus, Inc.



# Agenda

---

Rational Suite TestStudio

Protocols

SMTP

Writing Functions

POP3

VoIP

Questions



# Rational Suite TestStudio

---

GUI

VU

VU Recording

- API
- Network
- Proxy



# Protocols - Supported

---

TCP/IP Socket

HTTP / HTTPS

DBLIB

DCOM

IIOP

Jolt

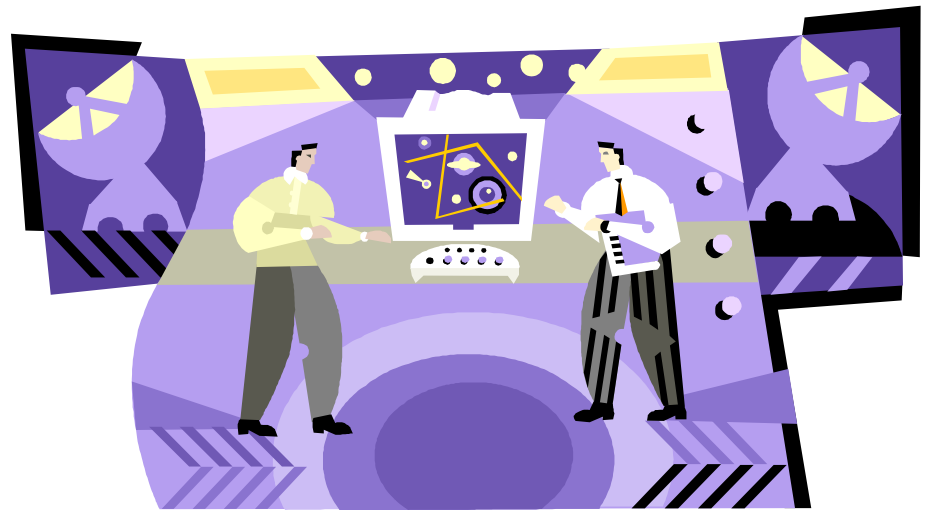
ODBC

Oracle

SQL Server

Sybase

Tuxedo



# Protocols - Unrecognized

---

## Recording Steps

- Network Recording
- Manual Filtering

## Examples of Protocols

- FTP
- SNMP
- SMTP
- POP3
- WAP
- NNTP



# SMTP – Why to Test

---

## Mail Server Testing

- Performance
- Redirects
- Mail Bombs
- Security



# Simple Mail Transport Protocol (SMTP)

---

## Port

- Default SMTP Port: 25

## Commands

- HELO
- MAIL FROM
- RCPT TO
- DATA
- RSET

## Formats

- Plain
- HTML

```
#include <VU.h>
int n;
{
Dmail_yahoo_com = sock_connect("SMTP001", "mail.yahoo.com:25");

set Server_connection = Dmail_yahoo_com;
n = sock_isinput();
sock_nrecv ["SMTP002"] n;

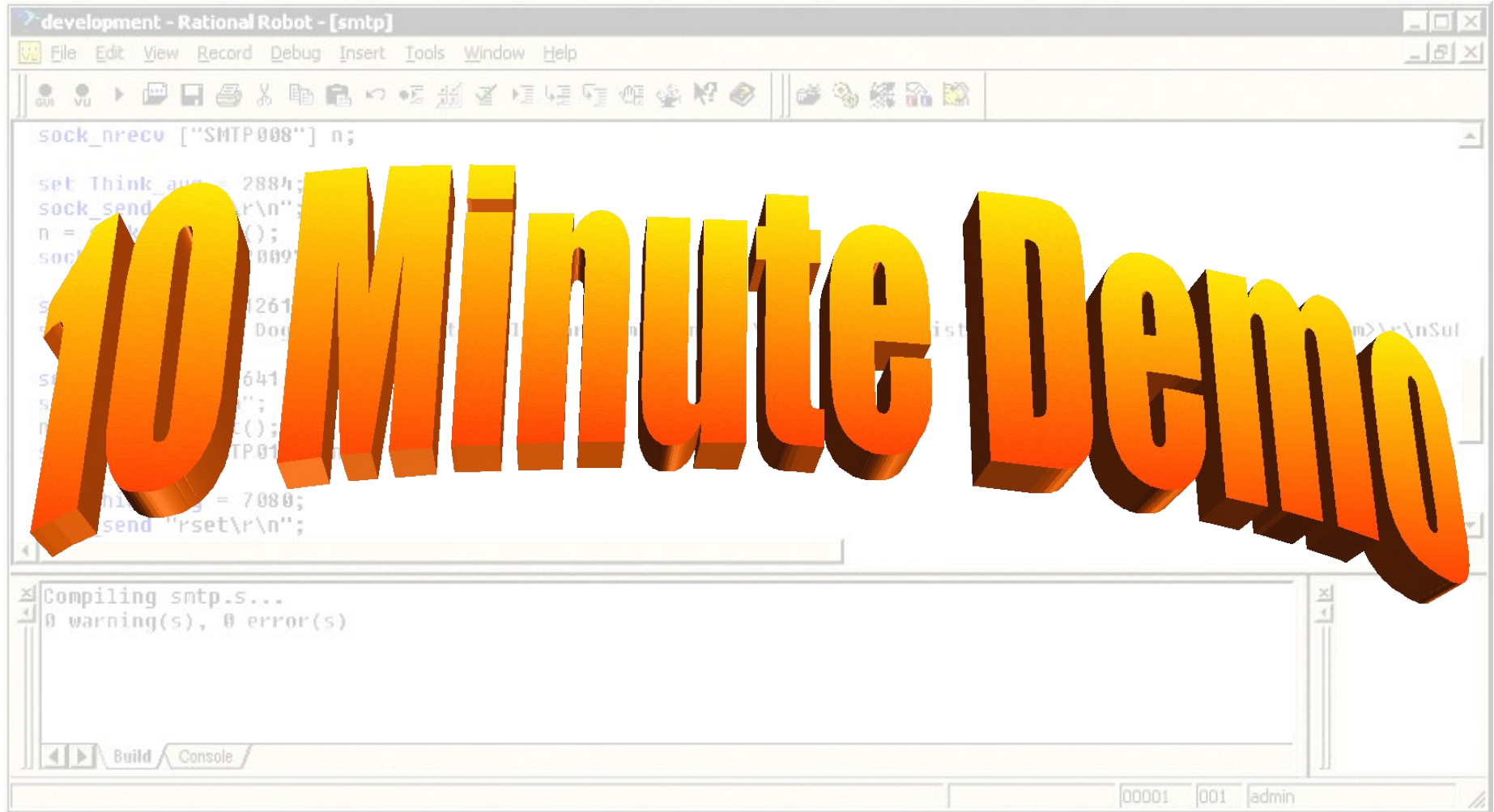
sock_send "helo john.doe.com\r\n";
n = sock_isinput();
sock_nrecv ["SMTP003"] n;

sock_send "mail from: dogbert@noblestar.com\r\n";
n = sock_isinput();
sock_nrecv ["SMTP006"] n;

sock_send "rcpt to: cwalters@noblestar.com\r\n";
n = sock_isinput();
sock_nrecv ["SMTP008"] n;
```



# DEMO - Recording Sending Mail



**10 Minute Demo**





# Writing Functions, Headers & Library Files

---

## Includes

- .sbh – SQABasic Header File
- .h – VUc Header File

## Constants

## Custom Functions

- .sbl – SQABasic Library File
- .h – VUc Library File

## Using Custom Functions in your script

- GUI
- VU



# Writing Functions - Example

---

```
#ifndef _SMTP_ // prevent duplicate inclusion
#define _SMTP_
int bytes;

int func smtp_helo(ip) // Initializes location mail is coming from
string ip; // IP address of mail originator
{   sock_send "helo " + ip + "\r\n";
    bytes = sock_isinput();
    sock_nrecv ["Smpt_Rcpt" + strep('0', 4 - strlen(itoa(_cmdcnt))) + itoa(_cmdcnt)] bytes;
}
int func smtp_from(sender) // Sets who the mail is from
string sender; // Email address of sender
{   sock_send "mail from: " + sender + "\r\n";
    bytes = sock_isinput();
    sock_nrecv ["Smpt_From" + strep('0', 4 - strlen(itoa(_cmdcnt))) + itoa(_cmdcnt)] bytes;
}
int func smtp_rcpt(receiver) // Sets who the mail is sent to
string receiver; // Email address of receiver
{   sock_send "rcpt to: " + receiver + "\r\n";
    bytes = sock_isinput();
    sock_nrecv ["Smpt_Rcpt" + strep('0', 4 - strlen(itoa(_cmdcnt))) + itoa(_cmdcnt)] bytes;
}
int func smtp_data(from, subject, body) // Creates the body of the mail
string from, subject, body;
{   sock_send "From: " + from + "\r\nSubject: " + subject + "\r\n" + body + "\r\n.\r\n";
    bytes = sock_isinput();
    sock_nrecv ["Smpt_Data" + strep('0', 4 - strlen(itoa(_cmdcnt))) + itoa(_cmdcnt)] bytes;
}
#endif // matches #ifndef _SMTP_ at top
```



# POP3 - Mail Server Test Suite

---

## Test Retrieval

- As an Employee
- External Mail vs Internal Mail
- As an Administrator

## Test Mail with a Virus

## Performance Test Mail Retrieval



# Post Office Protocol (POP3)

---

## Port

- Default POP3 Port: 110

## Login

- USER
- PASS

## Commands

- LIST
- RETR
- DELE
- STAT
- RSET

```
#include <VU.h>
int n;
{
Dmail_yahoo_com = sock_connect("SMTP001", "mail.yahoo.com:110");

set Server_connection = Dmail_yahoo_com;
n = sock_isinput();
sock_nrecv ["POP002"] n;

sock_send "USER cwalters\r\n";
n = sock_isinput();
sock_nrecv ["POP003"] n;

sock_send "PASS password\r\n";
n = sock_isinput();
sock_nrecv ["POP006"] n;

sock_send "LIST\r\n";
n = sock_isinput();
sock_nrecv ["POP008"] n;
```



# DEMO - Record Getting Mail

The screenshot shows the Rational TestManager interface. At the top, the title bar reads 'development - Rational TestManager'. Below it is a menu bar with 'File', 'View', 'Monitor', 'Tools', 'Window', and 'Help'. A status bar displays 'Testers: 1', 'Active: 0', 'Suspended: 0', 'Terminated: 0', 'Normal: 0', 'Abnormal: 0', and 'Time in Run: 00:00:44'. A toolbar contains various icons for test execution and monitoring. The main workspace is partially obscured by a large, 3D, orange-to-yellow gradient watermark that reads '10 Minute Demo'. Below the watermark, a table shows test results with columns for 'Iteration' and 'Users Inside'. At the bottom, a 'User View - Full' table is visible.

Groups		Script	Command	State	Time	Source		Cmd Count	Streak
Suite	Computer					File	Line		
1	User:Group1[1]	Local:computer[1]		Not Started			0	0	None



# Voice over IP (VoIP)

---

Protocols that VoIP works with

- TCP
- UDP
- IPv6

Protocol Format - Binary



# VoIP - Placing a Call

---

## Examples of VoIP Calls

- **Net2Phone**
  - Allows Internet to Phone or Internet calls
- **DialPad**
  - Allows Internet to Phone calls
- **Cisco IP**
  - Cisco Phone to standard Phone calls
- **Lucent**
  - Lucent Phone to standard Phone calls



# VoIP - Decoding Socket

---

## Protocol Headers

- Rational Suite TestStudio handles construction of the TCP/IP header

## Hexadecimal Data

- Protocol transmission is in binary format, look up RFCs to help decode

## Variable Determination

- Record multiple scripts with slight variations, like calling different numbers, do determine variables





# VoIP - Scripting

---

## Dialing

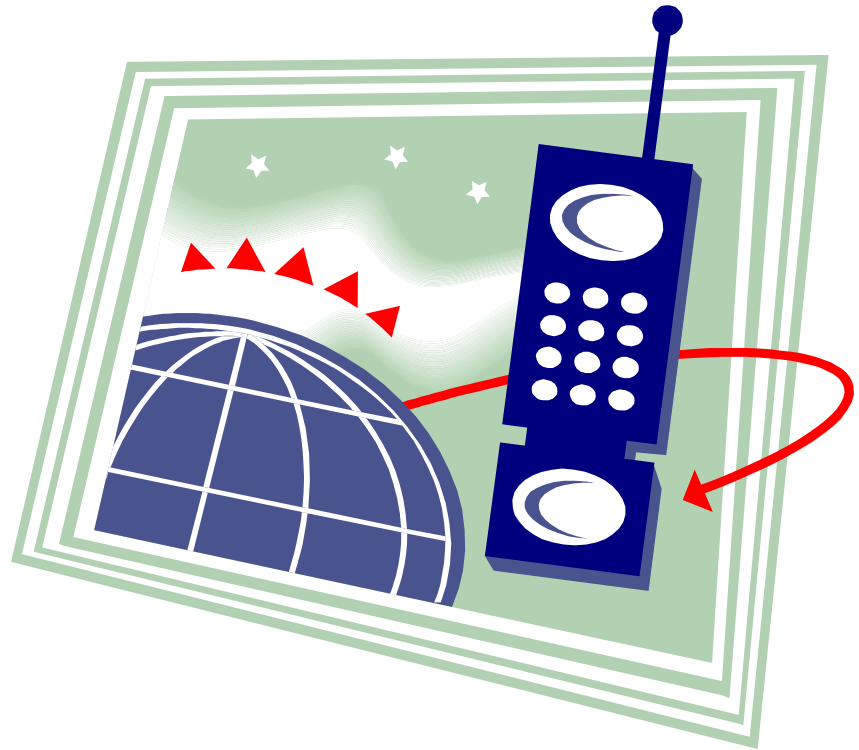
- ET Phone Home

## Data Transmittal

- Say "Hi"

## Hangup

- End Communication



# DEMO - Calling my Phone

The screenshot shows the Rational TestManager interface. At the top, the title bar reads 'development - Rational TestManager'. Below it is a menu bar with 'File', 'View', 'Monitor', 'Tools', 'Window', and 'Help'. A status bar displays 'Testers: 1', 'Active: 0', 'Suspended: 0', 'Terminated: 0', 'Normal: 0', 'Abnormal: 0', 'Time in Run: 00:00:44', and '0%'. A toolbar contains various icons for test execution and monitoring. The main workspace is partially obscured by a large, 3D, orange-to-yellow gradient watermark that reads '10 Minute Demo'. Below the watermark, a table shows test results with columns for 'Iteration' and 'Users Inside'. At the bottom, a 'User View - Full' table is visible.

Groups		Script	Command	State	Time	Source		Cmd Count	Streak
Suite	Computer					File	Line		
1	User:Group1[1]	Local:computer[1]		Not Started			0	0	None



# Conclusion

---

Rational TestStudio can record any TCP/IP Protocol

Functions make scripting unsupported protocols easy

- Writing custom functions requires:

- Knowledge of ANSI C
- Intimate knowledge of the protocol OR
- Access to someone else with intimate knowledge of the protocol

Custom Functions allow for testing of new, rare, and cutting edge technology without needing new tools or methodologies

---



---

# Questions?



# Contact Info

---

**Scott Barber**

*Chief Technology Officer*

*PerfTestPlus, Inc*

*E-mail:*

[sbarber@perftestplus.com](mailto:sbarber@perftestplus.com)

*Web Site:*

[www.PerfTestPlus.com](http://www.PerfTestPlus.com)

