Foundstone®
STRATEGIC SECURITY

WHITE PAPER

Auditing passwords using FSCrack

by Foundstone, a division of McAfee Inc®.

April 2006

Introduction

FSCrack is a front end for Solar Designer's John the Ripper (JtR). FSCrack provides an easy-to-use, intuitive graphical interface to access John the Ripper's main functions. John the Ripper is the de facto standard password cracker for both UNIX and Microsoft® Windows® operating systems passwords. While JtR is highly ranked among password crackers, it is only available as a command line interface (CLI) application. Not all users are comfortable using CLI applications, of course. If a given application is available in CLI or with a graphical user interface (GUI), most end users would opt for the GUI version. FSCrack was developed to bring the power of John the Ripper to end users. Now more users can perform security audits on their passwords and help secure their systems using one of the most popular freely available password audit tools.

This white paper provides background on FSCrack and the motivation for creating it, along with some common uses of the tool and sample screenshots.

Background and Motivation

What is John the Ripper?

From http://openwall.org/john :
"John the Ripper is a fast password cracker currently available for many flavors of Unix (eleven are officially supported, not counting different architectures), DOS, Win32, BeOS, and OpenVMS. Its primary purpose is to detect weak Unix passwords. several crypt(3) password hash types most commonly found on various Unix flavors, supported out of the box are Kerberos AFS and Windows NT/2000/XP/2003 LM hashes, plus several more with contributed patches."

What is FSCrack?

FSCrack is a GUI front end for John the Ripper, a password cracking program. FSCrack provides an intuitive interface for accessing its most common features. Instead of having to rely on the command line interface, users can now harness the power of JtR from a convenient GUI with a simple click of the mouse.

Why did Foundstone create FSCrack?

The idea for FSCrack was in part inspired by Nmapfe, a front end for Fyodor's Nmap. Of course, not all users are comfortable with command line interface. Many users prefer GUI-based applications over CLI-based applications due to their inherent ease of use. Nmapfe successfully brought the power of Nmap to GUI-oriented users.

In the spirit of bridging the gap for new or casual users, FSCrack was developed for those who prefer GUI applications over CLI applications. JtR is an incredibly powerful password cracker with a multitude of options. Its power and syntax might very well intimidate would-be users that do not feel comfortable with CLI applications. FSCrack does not add any new functionality to JtR; instead, it offers users a mechanism for building and executing JtR commands through a GUI.

Why should I audit my passwords?

During a round of "Ask the Author" posted on http://darwinmag.com, dated December 6, 2000, Bruce Schneier, the internationally renowned security technologist and author, was asked the question: "They say that security, like a chain, is only as strong as the weakest link. What is the most common 'weak security link' found in companies today?" His response was: "Stupid users."

What he meant is that users tend to create passwords that are easy to remember. Easy-to-remember passwords are also easy for others to guess or crack. For this reason, security administrators should audit user passwords within their organization as a way of testing their security.

Even with certain enforceable policies—such as minimum length, use of special characters, use of mixed case characters, etc.—most users can still figure out ways to create easy-to-remember passwords, which are usually also easy to crack. Many users are creative when attempting to dodge password creation policies. For example, they often replace letters with digits (e.g. "4" instead of "A", "3" instead of "E", etc.). Unfortunately for them, password crackers have also evolved. One of the very best (and free) available password crackers is Solar Designer's John the Ripper, which can cycle through dictionary words, making the same letter substitutions that users normally make when creating their passwords.

The case for stronger passwords

Most applications and systems today use weak authentication, requiring only a username and a password for accessing the operating system or a web application. One of the problems with this is that the word “password” is frequently taken literally. Often a user will simply use a dictionary word or a word that is meaningful to the user as a password. Easy-to-guess usernames and passwords represent a potentially high-risk vulnerability in systems and applications. Sometimes all it takes is one weak username and password to compromise an entire system and possibly an entire network. Even if a weak password does not lead to network or system compromise, it may be enough to allow an attacker to gain access to sensitive information, which, if disclosed, could lead to lost revenue, damaged reputation, and cleanup costs.

Well aware of these possible consequences, company policies often state that a password must be of certain length and contain certain characteristics. These policies can be enforced by most modern operating systems and by many applications that are designed to check passwords upon initial entry. Nonetheless, for every system with enforced password policies, there are almost always some passwords that do not meet the requirements. Normally, this comes from administrators who are not subject to the rules; test accounts that were set up without regard for the policies; or new accounts for which the password has not been changed by the user. It is wise to audit passwords occasionally to make sure they all meet password policies and are not easy to crack.

What makes a good password, then? According to best practices, a password or passphrase should have the following characteristics:

- It is at least fifteen characters long—usually, longer is better.
- It contains both uppercase and lowercase characters.
- It contains a combination of letters, digits, and special characters.
- It does not contain the username
- It is not made up solely of one or more dictionary words
- It does not simply contain numbers swapped for vowels (“4” for “a” or “0” for “O”)

The general recommendation is that users create a password from the first letter of each word in a sentence or phrase that they can easily remember. Then, some substitutions can be made. For example, the phrase: “This is going to be a great password” could become the password T!g2b@GP .

In addition to adhering to password length and complexity requirements listed above, one of the most effective means of preventing password cracking attacks is to enforce account lockout for failed attempts. Of course, if an attacker has gained access to the password database via some other attack, the account lockout will not prevent the attacker from performing offline cracking (with a tool such as JtR). The attacked can always come back to log in with one or more of the cracked passwords on the previously compromised system or on other systems on the network that the attacker did not have access to prior to cracking out the passwords on the compromised system.

Many password crackers found on the market today come with default rules for auditing passwords. JtR is no different and features four different cracking modes. The first three are run sequentially by default. These cracking modes are:

- Single mode, which uses the password file to generate some candidate passwords, such as using the username as a password.
- Wordlist mode, which uses a wordlist and can use word-mangling rules to attempt to crack passwords
- Incremental mode, which brute forces passwords using statistical data
- External mode, which cracks passwords based on user defined pseudo-C functions.

John the Ripper is commonly used by security professionals as a tool to audit password strength. It is also commonly used as a hacker tool to crack passwords. If the security community and hackers are using it to "test" your password security, so should you—before your password is cracked and misused.

Getting FSCrack and John the Ripper

The following are required to run FSCrack:

- John the Ripper binary (win32) written by Solar Designer – This is the John the Ripper executable and support files.
    - Available at: http://www.openwall.com/john/
- .Net framework 2.0.
    - Available at: http://msdn.microsoft.com/netframework/downloads/updates/default.aspx
- (Optional) NTLM (MD4) hash support patch written by Olle Segerdahl – This is the patched version of John the Ripper, featuring support for NTLM (MD4-based) password

hashes. This patch is required if one would like to determine the case sensitivity of previously cracked passwords, assuming they are in NTLM format (available for Microsoft Windows NT/2000/XP), which is something that JtR does not do by default.

  - Available at: http://olle.nxs.se/software/john-ntlm/

Features of FScrack

FSCrack's features can be divided into three categories: options, command builder, and output.

- Configuration options for JtR
- John the Ripper command builder -- a text area to view the command that is passed to JtR.
- John the Ripper output viewer -- a text area to view the output of JtR.

Please refer to the *FSCrack User Guide* for more detailed information on FSCrack's features.

Common FSCrack scenarios

Cracking passwords using JtR's default settings

FSCrack's most common function is to use JtR's default setting to crack passwords. To configure FSCrack to use this default setting, only two items are required:

1. The path to the JtR executable
2. The path to a passwd file

Once these two paths are selected, the user can click on the **Crack!** button and FSCrack will run JtR using the default cracking mode, which consists of up to three different modes in the following sequence:

1. "single mode" cracking, which uses the default "single mode" cracking rules defined in the john.ini configuration file
2. "wordlist mode" cracking, which uses the default wordlist file, passwords.lst
3. "incremental mode" cracking, which uses default word mangling rules also defined in the john.ini configuration file

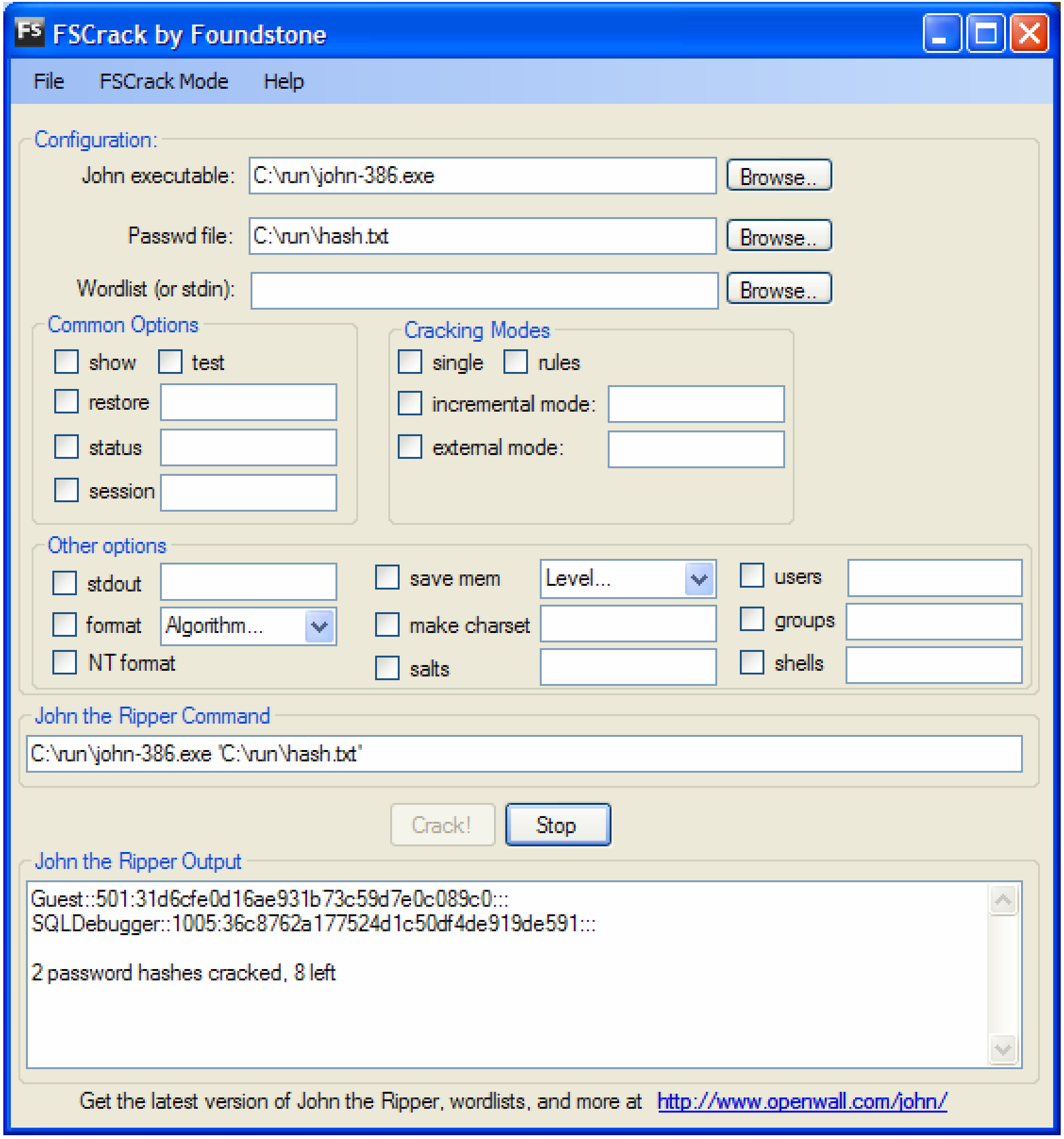
These three modes can be viewed as:

1. a quick check using very basic rules
2. another quick check using a basic password list
3. a brute force/hybrid attack.

For any casual user, this mode is the most effective and convenient method of running FSCrack and JtR.

This is the recommended approach for users when cracking passwords using JtR. The “single mode” and “incremental mode” are very time efficient. Casual users do not need to write their own rules for these two modes. The only recommended cracking mode that a casual user might want to customize is the “wordlist mode”. The “wordlist mode” takes any specified wordlist as input (selected using the **Browse** button to the right of the **wordlist file** label). If no wordlist is specified, JtR will use the default password.lst file. There are many free wordlists available on the internet. The JtR web site (http://openwall.org/john) also contains many different wordlists, available at a nominal fee.

This is a screenshot of FSCrack actively cracking password using the default configuration:



John executable: C:\run\john-386.exe

Passwd file: C:\run\hash.txt

Wordlist (or stdin):

John the Ripper Command

```
C:\run\john-386.exe 'C:\run\hash.txt'
```

John the Ripper Output

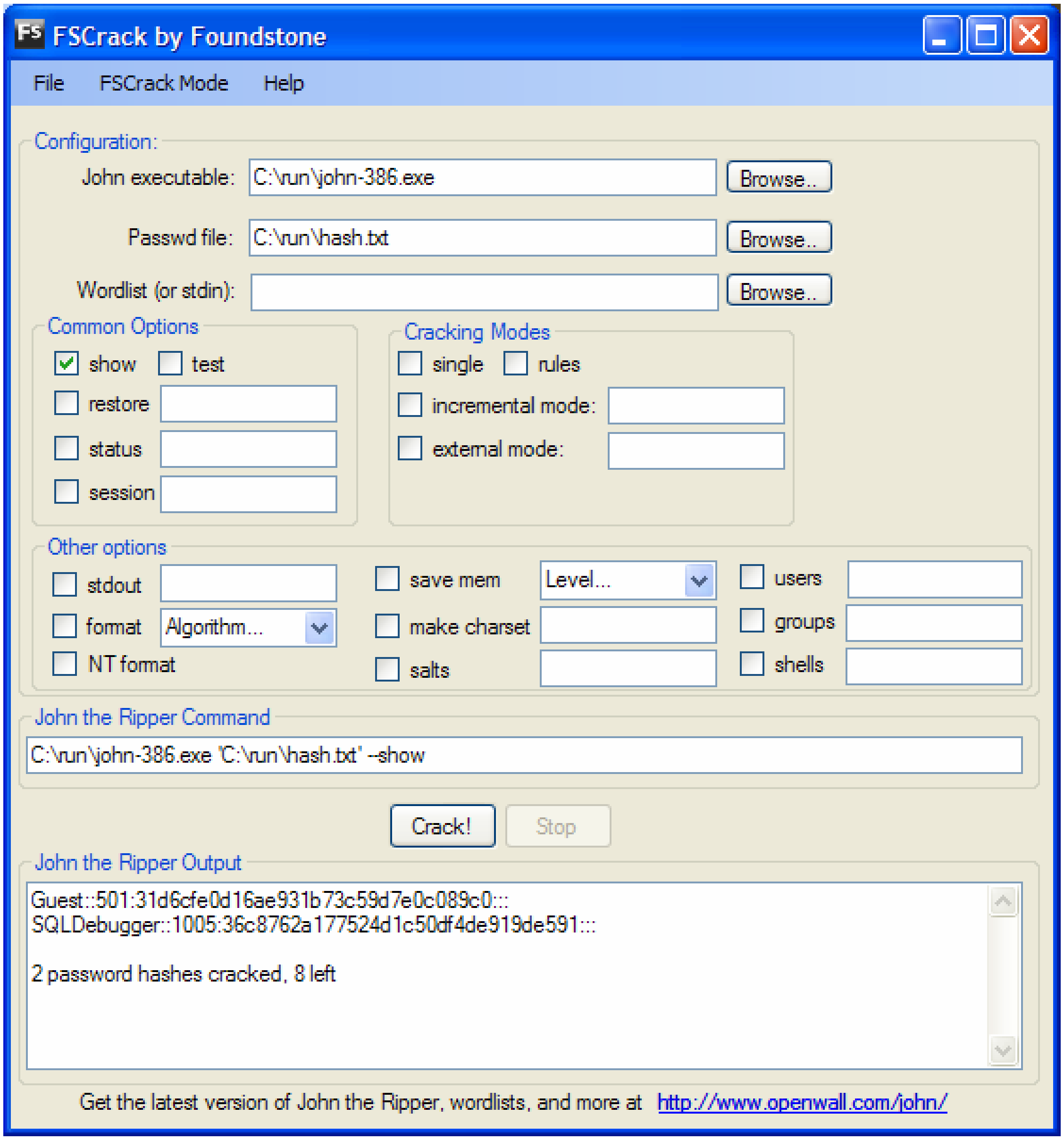
```
Guest::501:31d6cfe0d16ae931b73c59d7e0c089c0:::
SQLDebugger::1005:36c8762a177524d1c50df4de919de591:::

2 password hashes cracked, 8 left
```

Showing previously cracked passwords

The second most common use of FSCrack is to display previously cracked passwords. Using the default method of cracking—or any other method as detailed in Appendix A—cracked passwords can be displayed in the **John the Ripper output** text area using the **show** option.

The **show** option requires a passwd file. Once the **John executable**, **passwd file**, and **show** option have been selected, clicking on the **Crack!** Button will display cracked passwords for the specified passwd file.

The following is a screenshot of an end user running FSCrack with the **show** option to display previously cracked passwords from the file “c:\run\hash.txt



John executable: C:\run\john-386.exe

Passwd file: C:\run\hash.txt

Wordlist (or stdin):

☑ show ☐ test

☐ restore

☐ status

☐ session

☐ single ☐ rules

☐ incremental mode:

☐ external mode:

☐ stdout

☐ format Algorithm...

☐ NT format

☐ save mem Level...

☐ make charset

☐ salts

☐ users

☐ groups

☐ shells

```
C:\run\john-386.exe 'C:\run\hash.txt' --show
```

```
Guest::501:31d6cfe0d16ae931b73c59d7e0c089c0:::
SQLDebugger::1005:36c8762a177524d1c50df4de919de591:::

2 password hashes cracked, 8 left
```

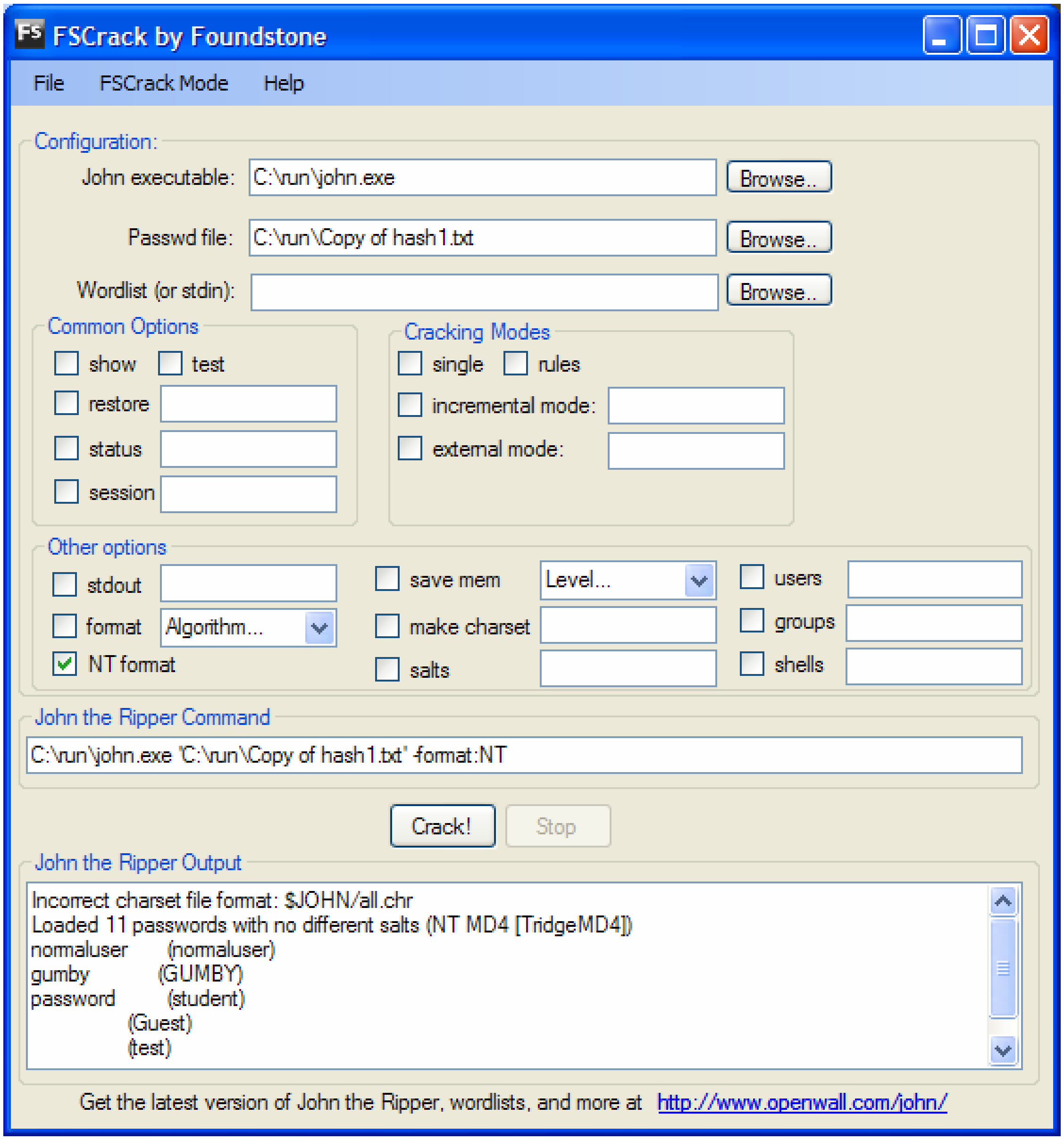
Get the latest version of John the Ripper, wordlists, and more at http://www.openwall.com/john/

Displaying NTLM (MD4) passwords

Once you have cracked Microsoft Windows passwords, you may want to figure out the case sensitivity of the cracked passwords. You can do this using the patch provided by Olle Segerdahl, vailable at: http://olle.nxs.se/software/john-ntlm/. After you download the patched version of JtR, select this patched version as the **John executable**, select the **passwd file** to view the

NTLM values for, select the **NT format** option, and click on the **Crack!** button. FSCrack will output any NTLM formatted passwords to the **John the Ripper** output text area.

The following is a screenshot of an end user running FSCrack with the **NT format** option to display the NTLM (MD4) value of the previously cracked passwords.



John executable: C:\run\john.exe

Passwd file: C:\run\Copy of hash1.txt

Wordlist (or stdin):

☑ NT format

John the Ripper Command:

```
C:\run\john.exe 'C:\run\Copy of hash1.txt' -format:NT
```

John the Ripper Output:

```
Incorrect charset file format: $JOHN/all.chr
Loaded 11 passwords with no different salts (NT MD4 [TridgeMD4])
normaluser      (normaluser)
gumby          (GUMBY)
password        (student)
                (Guest)
                (test)
```

Get the latest version of John the Ripper, wordlists, and more at http://www.openwall.com/john/

Known issues

There are no known issues at this time.

Acknowledgements

A special thanks to Solar Designer (et al.) for creating such a great password cracker. Also, a special thanks to all who contributed patches to JtR.

About Foundstone Professional Services

Foundstone Professional Services, a division of McAfee, Inc. ®, offers a unique combination of services and education to help organizations continuously and measurably protect the most important assets from the most critical threats. Through a strategic approach to security, Foundstone identifies, recommends, and implements the right balance of technology, people, and process to manage digital risk and leverage security investments more effectively.