



# ***Information System Security Operation***

## **Community-Based Open Source Security**

### **Improved Security of Open Source Software, with a Focus on the FreeBSD Operating System**

#### **Problem**

The security of Open Source software, such as FreeBSD, has become critical to the United States information infrastructure. Tens of thousands of Government and commercial organizations depend on Open Source software, and this dependence is increasing. Unfortunately, this software is not secure, and almost every day the moderated mailing list *BugTraq*, CERT®, and other venues report serious new security flaws.

Many “new” security flaws are actually old vulnerabilities that have existed in the code, unknown, for years. Open Source code contains many vulnerabilities, and additional vulnerabilities are added continuously as new source code is generated. There is no “silver bullet” solution, however, there are compelling opportunities for improving the way Open Source code is generated, and thus the security of Open Source software.

#### **Solution**

FreeBSD is an Open Source, advanced, high-performance operating system widely used by internet service providers, and is the basis for embedded network products including routers and firewalls, due to its scalability, ease of management, and cost effective operation. The system has also formed the basis for substantial parts of Apple's Mac OS X next generation operating system, as well as products from other operating system vendors.

U.S. Navy's Space and Warfare Systems Command awarded a contract to McAfee® Research, now the Security Research Division at SPARTA, to develop security extensions to FreeBSD. That contract is for the Composable High Assurance Trusted Systems (CHATS)

Program. It is funded under the Defense Advanced Research Projects Agency (DARPA). We have conducted the CHATS Program in partnership with members of the FreeBSD developer community, assuring tight system integration and rapid technology transfer.

The CHATS Program has brought together key elements of the Open Source and computer security communities to measurably improve real-world security of critical Open Source systems. The Community-Based Open Source Security (CBOSS) effort has implemented a strategy based on the deployment of highly general security architectures and effective community participation in technology transfer to initiate meaningful and lasting changes in the way the Open Source community develops software.

#### **Approach**

We have assembled a team of Open Source leaders and operating system security experts and has initiated a self-sustaining security infrastructure within the Open Source community. With the participation of several of the Open Source community's most influential members, the CBOSS effort is transferring the knowledge, tools, and techniques required to enable the community to routinely develop software that is sufficiently secure to support the National Information Infrastructure.

The CHATS program at DARPA focuses on the development of the tools and technology that enable the core systems and network services to protect themselves from the introduction and execution of malicious code and other attack techniques and methods. These tools and technologies have provided the high assurance trusted operating systems the security services needed to achieve comprehensive secure highly

---

This work sponsored by DARPA through SPAWAR,  
Contract Number N66001-01-C-8035.



## Community-Based Open Source Security

Improved Security of Open Source Software, with a Focus on the FreeBSD Operating System

distributed mission-critical information systems for the DoD.

The team has identified key community-based initiatives that, in combination, address the most important current deficiencies:

Through the transfer of existing security knowledge initiative, we have forged a partnership between the computer security research community and selected Open Source developers. Our work has provided architectural guidance, and has also focused strongly on providing practical nuts-and-bolts security information to programmers. We have developed an increased awareness of security among Open Source developers, and set in motion a “virtuous cycle” of security improvement in Open Source projects.

In close collaboration with Open Source developers, the initiative to transfer existing security technology has implemented a collection of advanced security technologies (operating system kernel improvements, authentication improvements, etc.), and work to ensure that these technologies are incorporated into mainstream systems, such as FreeBSD. By quickly applying advanced, but poorly deployed security technology, we have produced immediate benefits to Open Source systems. Perhaps more importantly, these efforts have fed into our first initiative (Transfer of Existing Security Knowledge) by serving as worked examples of how security knowledge can be captured and communicated in the context of “running code.”

The initiative for producing kernel security extensions has developed new technology for adding security features to operating system kernels. This capability is critical because new security features are often required, but cannot be realistically added because current operating system kernels provide little or no support for security extensibility. In particular, current systems provide no support for policy composition. This research has provided such extensibility, thus harnessing traditional Open Source development processes (quick experimentation, competition among features and implementations) to develop and validate

effective kernel-based security features and policies.

As with our other initiatives, this initiative has focused on engaging the Open Source community, and on setting in place a process that will continue long after DARPA’s direct investments have ended.

The new technology initiative for producing high-security applications has developed a pragmatic technology for structuring (or restructuring) UNIX application programs (e.g., WU-FTP, Apache, BIND) that are effectively immune from a central malady that continues to plague Open Source programs: root compromise through buffer overflow attacks, format bugs, misconfiguration, or design error. This initiative has demonstrated worked examples, and has shown how the techniques can be used in existing and new Open Source projects. Through a community-based process, we have changed the default structure of UNIX software, and have eliminated the debilitating class of attacks on Open Source systems.

We believe that these initiatives have targeted the “low hanging fruit” in Open Source system security, and prepared for longer-term research aimed at more revolutionary change.

Many members of our team are well-known members of the Open Source community and are in a position to ensure that the community as a whole seriously considers our work.

Through our community-based approach, we seek to provide the following key benefits:

- Prevention of many future security flaws through increased security knowledge.
- Elimination of a number of critical security vulnerabilities in existing code.
- Dramatically improved kernel security support.
- Dramatic reduction of “get root” attacks on critical Open Source applications.