

Bishop_P033_TXT.txt

>Type of Submission: Paper
>Title: A Vector-Based Approach To Vulnerabilities Analysis
>Topic Category: Vulnerabilities and Attacks -- Vulnerability
> or attack taxonomie
>Name of Speaker: Matt Bishop
> (phone) (530) 752-8060
> (fax) (530) 752-4767
> (email) bishop@cs.ucdavis.edu
> (postal) Department of Computer Science
> University of California at Davis
> Davis, CA 95616-8562
> USA
>Brief Biography:
> Matt Bishop received his Ph.D. in computer science from Purdue
> University, where he specialized in computer security, in 1984.
He
> was a research scientist at the Research Institute of Advanced Co
mputer
> Science and was on the faculty at Dartmouth College before joinin
g the
> Department of Computer Science at the University of California at
> Davis. His research areas include computer and network security,
> especially analysis of vulnerabilities, building tools to detect
> vulnerabilities, and ameliorating or eliminating them.
>Subject Category: Vulnerabilities and Attacks
>
>
>Extended Abstract:
>
>The goal of our work is to develop a methodology for detecting pre
viously
>unknown vulnerabilities in systems.
>
>We begin by arguing that existing classification schemes are inade
quate
>because they either lump vulnerabilities together at a high level,
or
>fail to capture shared characteristics of vulnerabilities. For exa
mple,
>race conditions caused by file accesses and race conditions caused
by
>simultaneous signals can be classified identically using PA and RI
SOS.
>But the two are fundamentally different. Similarly, race conditio
ns

Bishop_P033_TXT.txt

>involving file accesses and vulnerabilities arising from improper system
>configuration are classified differently using Aslam's and Krsul's
>classification scheme. The two share a common characteristic and, while
>different, are related.
>
>We define the following criteria:
>
>1. Similar vulnerabilities are classified similarly. However, we
>do not require that they be distinct from other vulnerabilities.
>Because a vulnerability can rarely be characterized in exactly one way,
>a realistic classification scheme must take the multiple characteristics
>causing vulnerabilities into account. This allows some structural redundancy
>in that different vulnerabilities may lie in the same class; but
>we expect (and indeed desire) this overlap.
>
>2. Classifications are primitive, i.e., each class has exactly one property.
>
>3. Classification terms should be well-defined. We analyze this
>at length in our paper and explain approaches (such as using thesauri).
>
>4. Classification should be based on the code, environment, or other technical details. For our purposes, social causes of vulnerabilities are irrelevant.
>
>We represent a vulnerability as a set of characteristics.
>Characteristics may be determined formally when a formal top-level specification is present, or empirically when only general security properties are known. The paper will give several examples of characteristics. We will discuss two specific vulnerabilities (one a race condition, the other a buffer overflow) and contrast this scheme with the other ones named above.
>
>For example, an exploitable race condition involving file accesses has two characteristics: a window in the code, and a name in the code that can have its binding to an object changed. Thus, it shares common characteristics with vulnerabilities involving making system

>directories world readable.
>
>We present techniques for deriving informal characteristics, and
>discuss minimality, soundness and completeness requirements:
>
>* A minimal set has the properties that (a) it describes a vulnera
bility,
> and (b) if any characteristic in the set does not hold, the vuln
erability
> no longer exists.
>
>* A sound set has the property that characteristics in the set do
not
> overlap.
>
>* A complete set has the property that all vulnerabilities of a sy
stem
> have characteristics drawn from this set.
>
>We discuss these sets in systems that have formal top-level specif
ications
>as well as those which do not.
>
>We hypothesize that every vulnerability has a
>a minimal, sound set of sound characteristics. We will provide
>evidence to support our hypothesis. We derive properties
>of vulnerabilities such as distance that are useful in a classific
ation,
>and show how such properties aid us in our search for unknown
>vulnerabilities.
>
>Characteristics are central to this scheme meeting its goal.
>Specifically, we wish to test the following:
>
>* We can determine the characteristic set for any vulnerability
>
>* The size of a complete set of characteristics for a system is
>significantly smaller than the size of the set of vulnerabilities
>
>* Each characteristic suggests a tool to analyze the system or
>system programs to determine if the condition exists.
>
>We examine the reasonableness of these claims in the context of
>race conditions, buffer overflows, and IP spoofing on the Interne
t.
>We discuss building tools to look for characteristics. We describ

Bishop_P033_TXT.txt

e

>several previously unknown vulnerabilities found using such a tool

.

>We demonstrate that ancillary results (such as vulnerability

>databases and thesauri) will be useful in their own right.

>

>