Vulnerability Analysis

Vulnerability analysis at the CERT Coordination Center (CERT/CC) consists of a variety of efforts, with primary focus on coordinating vulnerability disclosure and developing vulnerability discovery tools and techniques. Publicly available resources include:

- Public vulnerability information: Vulnerability Notes and vulnerability data archive
- Coordination and disclosure guidance for security researchers and vendors
- The CERT Guide to Coordinated Vulnerability Disclosure in its entirety
- Vulnerability Disclosure Policy Templates for use in creating your own customized disclosure policy
- Vulnerability Reporting Form (please be familiar with the guidelines before reporting)
- Open-source vulnerability discovery and analysis tools
    - CERT BFF - Basic Fuzzing Framework — The CERT Basic Fuzzing Framework (BFF) is a software testing tool that finds defects in applications that run on the Linux and Mac OS X platforms. BFF performs mutational fuzzing on software that consumes file input.
    - CERT FOE - Failure Observation Engine — The CERT Failure Observation Engine (FOE) is a software testing tool that finds defects in applications that run on the Windows platform. FOE performs mutational fuzzing on software that consumes file input.
    - CERT Tapioca — CERT Tapioca is a network-layer man-in-the-middle (MITM) proxy framework based on mitmproxy http://mitmproxy.org/. CERT Tapioca is installable on Red Hat Enterprise Linux, CentOS, Fedora, Ubuntu, OpenSUSE, and Raspbian.
    - CERT Triage Tools — The CERT Triage Tools project has been transitioned to the GDB 'exploitable' plugin https://github.com/jfoote/exploitable project on GitHub.
    - CERT Vulnerability Data Archive and Tools — The CERT Vulnerability Data Archive contains nearly all of the non-sensitive vulnerability data collected by the CERT/CC, from the inception of the vulnerability notes database (approximately May 1998) to the date the archive was prepared, as noted above in the Change Log.
    - Dranzer — Dranzer is a tool that enables users to examine effective techniques for fuzz testing ActiveX controls.