This letter provides information about the limited vulnerability of HP Integrity OpenVMS systems and software virus attacks on the system.

HP OpenVMS Security

HP OpenVMS is a highly available and secure operating system that is easy to use. HP OpenVMS is most popular in environments where stability, reliability, high performance, disaster tolerance, security, scalability and clustering are required.

- OpenVMS is secure out-of-the-box, as we ensure that very few default accounts are created, the few that are created are either disabled or operate with restricted access. All unnecessary services are also disabled by default. We also implement a robust access control mechanism that allows for granting fine grained access to end users on a need basis.
- The design of the OpenVMS operating system uses four modes of protection – kernel (the innermost and most privileged mode), executive, supervisor, and user (the outermost and least privileged mode) -- enforced by the hardware.
- Strings are passed to OpenVMS system routines using a "string descriptor", an OpenVMS construct that includes elements like the data pointer and length of the data. This is a very powerful construct that eliminates buffer overflows, one of the most common sources of vulnerabilities.
- While there is confusion over which products and services are necessary, certain infrastructure components should protect every system. Infrastructure should include an enterprise security policy, Safeguard, and additional measures such as proxy servers and/or firewalls to protect against threats like denial-of-service attacks

HP OpenVMS CRTL Security

It may be useful in this discussion to describe a few distinctions about the POSIX compliant C Runtime Library (CRTL) implemented on OpenVMS. While CRTL is POSIX compliant, CRTL is not POSIX, UNIX, or LINUX. All of the code used in CRTL, including the low-level kernel code, was implemented by HP's OpenVMS Engineering Team to our own software engineering standards and therefore inherits the OpenVMS Operating System fundamentals.

This is important because non-privileged (user) processes running on OpenVMS cannot:

- corrupt the code space of another process running on the OpenVMS System; or
- corrupt the data space of another process running on the processor unless the two processes agree in advance to share memory, and even then only the shared memory could be compromised; or
- escalate their set of privileges to gain unauthorized access to code or data

Thus, applications written by a customer in accordance with HP guidelines and recommendations will achieve the security and availability benefits of CRTL running on HP's OpenVMS Operating System.

Viruses

With respect to your concerns about virus susceptibility, this discussion uses the definition from the Microsoft Dictionary: a virus is "an intrusive program that infects computer files by inserting in those files copies of itself. The copies are usually executed when the file is loaded into memory, allowing the virus to infect still other files, and so on. Viruses often have damaging side effects-sometimes intentionally, sometimes not. For example, some viruses can destroy a computer's hard disc or take up memory space that could otherwise be used by [the] program."

As long as an OpenVMS customer has properly secured their OpenVMS system, object files, including executable and shareable images, cannot be written to by a non-privileged process. This means that even if a virus written for a UNIX, LINUX, or POSIX environment on any other platform were to be introduced to an OpenVMS system, it could not execute. That is, binary (object) code that could bring down a UNIX environment running on a different vendor's system could not execute on an OpenVMS system. Common UNIX attacks that exploit buffer overflows to gain root access capabilities will not work on an OpenVMS System not only because such attacks rely on a specific processor type, but also because a non-privileged process cannot escalate its set of privileges. That said, there is an important caveat to keep in mind: it is possible that a virus could be propagated by common software that has been ported to OpenVMS, such as SAMBA or a POP or SMTP mail server even though the OpenVMS System itself would not be impacted by the virus. This issue can be addressed by deploying software designed to scan for non-native viruses on OpenVMS systems acting as file or email servers.

Any virus that could replicate itself on an OpenVMS System would have to be specially designed. Crafting an appropriate virus, which could only work for a single version of a single program at a time, requires high level access to the code on the OpenVMS Server. But that is not a weakness in the OpenVMS system, it is exploitable only if a customer fails to adequately protect its source code and systems from unauthorized or hostile access and fails to use safe operating techniques.

It is the OpenVMS customer's responsibility to ensure that all applicable security patches have been applied on the OpenVMS system to help minimize this risk. In fact, when it comes to security, most issues manifest themselves at the user level: opening attachments; using weak passwords; visiting websites that download 'spyware' and sharing sensitive information. These problems must be addressed by the customer at an operational level by ensuring good security practices such as separation of duties, assigning "least privilege" capabilities, controlling source code, and having appropriate software configuration management procedures.

In addition, the effects of attacks on networked systems can be minimized by the customer employing a combination of standard network protection techniques, an intrusion detection system (IDS), regular auditing and writing and using quality application code that contains a minimum of exploitable conditions.

Since 1989 (Father Christmas, Morris and WANK) no virus or worm has been reported to Hewlett-Packard, Compaq Computer Corporation, or Digital Equipment Corporation as existing on any OpenVMS system. That is not to say that such issues could never occur, but until now there is no history that could be used to create the signatures used by a "virus scanner" for HP OpenVMS. We

will continue to examine the need for a virus scanner for HP OpenVMS. The HP OpenVMS Enterprise division works with a specially tasked internal HP security group to quickly respond to reports of new vulnerabilities.

While there are no absolutes, the information in this letter is provided to help customers understand that HP takes security seriously and does what it can to ensure that customers are comfortable running their most critical and trusted applications on HP OpenVMS systems. Hopefully this letter has helped to explain why we believe the risk of virus attack on the HP OpenVMS system is very low.

Regards,

HP OpenVMS
Product Management team