**Module – 3 | Security in Computing**

**Risks**

* Buffer overflow
  + Often come from innocent programmer oversights or failures to document and check for excessive data.
  + Falls strictly within a data space or spills over into an adjacent code area.
* Data-driven attack
  + Overwrite stack memory, sometimes called stack smashing, in a purposeful manner.
  + Ways the attacker can utilize an overflow attack:
    - Overwrite the program counter stored in the stack so that when this routine exits, **control transfers to the address pointed at** by the modified program counter address.
    - Overwrite part of the code in low memory, **substituting the attacker’s instructions** for previous program statements.
    - Overwrite the program counter and data in the stack so that the program counter now **points into the stack**, causing the data overwritten into the stack to be executed.
* Countermeasures for above
  + Independent testing and code review.
  + Code analyzers.
  + Separation.
  + Time-Of-Check To Time-Of-Use
    - Between access check and use, data must be protected against change to avoid hijacking of the data for other purposes.
* Malicious code – Malware
  + Virus
    - Replicate itself and pass on malicious code to other non-malicious programs by modifying them.
  + Worm
    - Program that spreads copies of itself through a network.
  + Trojan horse
    - Malicious code that, in addition to its primary effect, has a second, nonobvious, malicious effect.
  + Bot (short for robot)
    - Kind of worm used by search engine hosts that scan accessible web content continuously and report back to their controller.
  + Transmission and Duplication
    - Setup and installer program transmission.
    - Attached file.
    - Document viruses.
    - Autorun
    - Propagation
    - Appended viruses.
    - Viruses that surround a program.
  + Homes for viruses
    - Embedding
    - One-time execution (implanting)
    - Boot-sector viruses
    - Memory-resident viruses
    - Application programs