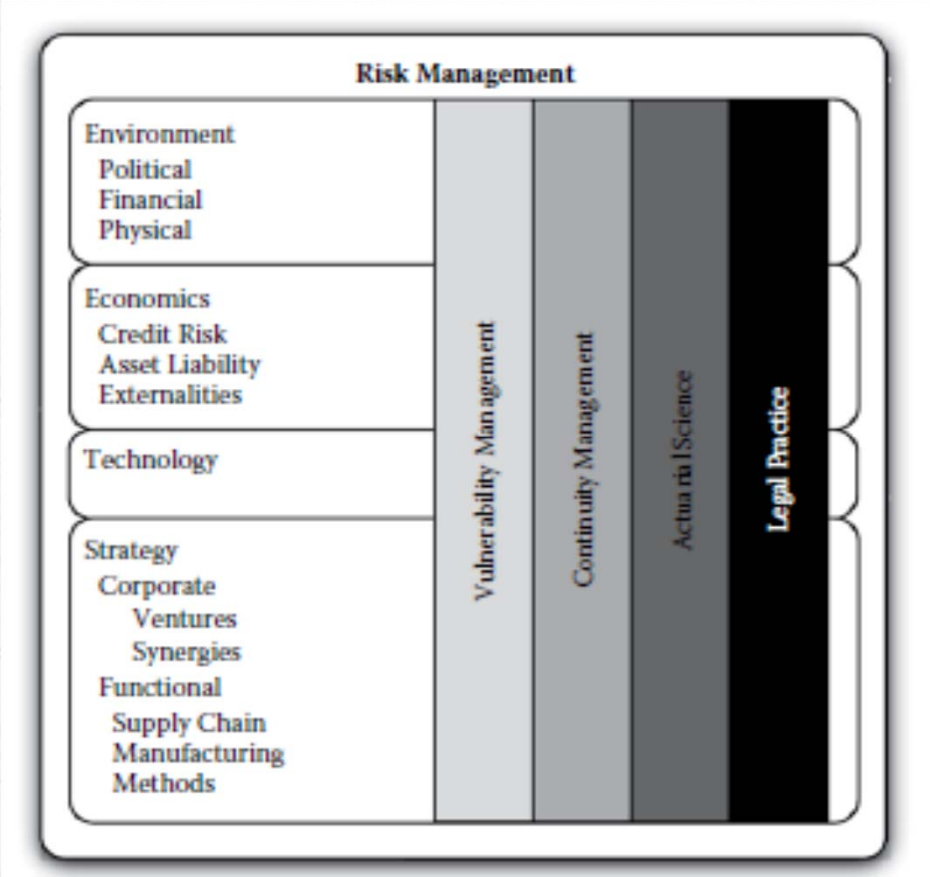
The Role of Risk Management

- Relationship of VM with a risk management program
- Discover and address risks that result from vulnerabilities
- VM typically focuses attention on technical software and system configuration vulnerabilities



The Role of Risk Management

- There are also vulnerabilities related to corporate strategy, economics, and the environment whose detection cannot be automated
- These vulnerabilities exist in areas such as business processes, strategies, and supply chains. Every action or plan that a business has could be exploited through design flaws or a lack of adaptability
- It is the larger role of the risk manager to recognize and address these challenges

Vulnerability Management and Risk Management

- The overall IT risk management framework is similar to vulnerability management
- IT risk management stages are to identify critical assets, identify and rank risks, identify controls, implement controls, and then monitor the controls' effectiveness

Table 1-1: Mapping Vulnerability Management to IT Risk Management

Vulnerability management	IT risk management
Collect data	Identify critical assets
Analyze data	Identify and rank risks
Make recommendations	Identify controls
Implement recommendations	Implement controls
(Collect data)	Monitor controls

Asset Information

- Many organizations, large and small, don't have a full—or even fragmentary—understanding of what is on their networks
- Need a complete inventory of IP-connected devices and any additional data that you can glean about each host
- Obtaining a list of hosts—and even a wealth of additional information—is straightforward
- You can use a network-scanning tool, like Nmap, or a vulnerability scanner, like Nessus or Qualys to do a network sweep and find live hosts
- But these scans can be obtrusive and might cause application or even OS crashes
- So, you need to carefully plan for an information-gathering scan

Asset Information

- New devices are added to networks all the time
- although most organizations have a change management policy in place
- There is no guarantee that changes aren't made without following policy
- To have updated and trustworthy asset information, you must perform discovery scans on a regular basis across the entire network

Vulnerability Information

- Configure a vulnerability scanner to do a deep scan of each device and discover any known host vulnerabilities
- Carefully look at the available scanner options and tailor the settings to your environment and risk tolerance
- Scan some of your network sections, such as endpoint segments, every day
- Risk of downtime is limited and the consequences aren't severe
- scanning your core production databases, might be too risky to do outside of scheduled maintenance windows
- Trade-off between getting fresh data and risking downtime

Vulnerability Information

- Network vulnerability scanners will only find vulnerabilities that are discoverable over a network connection
- A network scan won't find a locally exploitable vulnerability in a desktop application on a Windows endpoint
- For example, a network scanner won't find CVE-2018-0862—a vulnerability in Microsoft Equation Editor that an attacker can only exploit by opening a crafted Word or WordPad document
- The reason is that Microsoft Office applications in general aren't detectable via a network scan

Exploit Data

- Information about publicly available exploits is widely accessible and often searchable
  - https://www.exploit-db.com/
- Use CVE ID to correlate exploit information with vulnerability information that you already possess
- Addressing an exploitable vulnerability is likely a higher priority to your organization than a vulnerability that isn't yet known to be exploitable
- An exploit that enables arbitrary code execution is more severe than one that permits reading arbitrary data
- Knowing the consequences of an exploit is very useful for prioritizing exploits with more granularity

Advanced Data Sources

- **Threat intelligence feeds:**
  - Include information about the current threat landscape: threat actors and groups
  - The exploits currently being used in exploit kits, and the vulnerabilities with privately available exploits that aren't yet public knowledge
  - Use this information to determine which vulnerabilities are currently a higher risk to your organization
  - Because these threat feeds contain fresh data, you should use the feed data as soon as it comes in to get a timely assessment of your exposure to newly discovered threats
  - Numerous free and paid threat feeds are available, such as iSight Threat Intelligence, iDefense Threat Intelligence, and industry-specific threat feeds, like the one provided by FS-ISAC

Advanced Data Sources

- **Proprietary exploits:**
  - Sources range from commercial threat intelligence sources that commission their own exploit research to decidedly grey- or black-market options
  - such as independent researchers selling newly discovered vulnerability and exploit information to the highest bidder
  - Whatever the source, proprietary exploit information will help you better prioritize your own vulnerability data based on exploits you would otherwise be unaware of

Advanced Data Sources

- **Network configurations :**
  - Use network configurations from routing devices like routers, firewalls, and managed switches to create a model of your network
  - By combining this information (which subnets route to which, which ports are accessible from where) with vulnerability and exploit data, you get a deep understanding of your network attack surface
  - For example, if a Tomcat exploit exists for an internal web application server but your router configuration indicates that
  - This server is accessible only to a limited list of source IP addresses, it might be of less concern to you than if it were accessible to the internet at large
  - It takes significant work to integrate this data with your existing vulnerability data

Data source	Important data
Host/port scanner (Nmap)	IP address MAC address Hostname Open ports (TCP and UDP) Service and OS fingerprinting
Network vulnerability scanner	(Same as above) Additional service fingerprinting and version detection Network vulnerabilities Local vulnerabilities (authenticated scans only)
Host-based vulnerability scanner	Local vulnerabilities
CMDB/SCM	OS details Deployed software details Configuration details Owner of the device Criticality of the device and application
Exploit databases	Exploit information Vulnerability mapping to exploits
Threat intelligence	Attacker and targeted industry intelligence Newly discovered, escalating, or widespread exploits
Exploit kits	Proprietary exploit information
Network configurations	Network topology and potential attack paths

The Vulnerability Creation Process

- Should consider how programs are developed now a days:
- 1. Concept and proposal: A business or government operation has a required function
- 2. Requirements gathering: The developer analyses the requirements and clarifies them with the business
- 3. Design: A design is developed against the requirements
- 4. Code: Programmers write the code and test it
- 5. Test: The quality assurance team makes sure that the code meets the requirements

The Vulnerability Creation Process

- 6. Deployment: The business installs the software on production systems and announces it to the world. Or, the software is shrink-wrapped and shipped to stores everywhere
- At each step along the way, there are problems that result in missing something that users ultimately regret
- 7. Identification: A hacker, researcher, or end user finds a flaw in the processing of inputs, outputs, or interaction with other system functions. This could also be considered hidden or new functionality
- 8. Exploitation: A person then tries to see how many ways this flaw can be manipulated to exceed the original requirements of the software
- 9. Compromise: An additional piece of software is developed and deployed to recreate these circumstances and automate the new functionality

The Vulnerability Creation Process

- Three factors have made items 7 through 9 more commonplace and more problematic than ever: complexity, communication, and interoperability

- Complexity :
  - The environment and systems upon which these programs operate are enormously complex e.g. OS

- Communication :
  - Built-in security flaws in Internet protocols
  - Misapplication of a technology

- Interoperability :
  - All the system have the capability of interoperability as they have implemented the standard in their product by their way
  - Every vendor would have to identify and patch their products