In an attempt to get this started as a whole class, and I hope I’m not jumping the gun, but what I’ve thought would be nice was to see a 10 step process.

Here is what I have so far. Let me know your thoughts.

10 step VAM steps

1 - Identify essential information functions.

2 - Categorize and set Policy based on Information Systems role.

3 - Identify vulnerabilities of Information Systems.

4 - Implement Policy/Security Controls.

5 - Perform risk assessment on policies applied and systems being utilized.

6 - Map the network infrastructure that connects the hardware.

7 - Test the techniques applied in step 5 for robustness and actual feasibilities under threat.

8 - Apply fixes based on testing in step 7.

9 - Actively monitor Security controls and re-verify controls. Gain Threat intelligence.

10 - Spend time clearly defining and documenting measures taken, policies applied, and systems identified.

I think #1 should be Identify ALL information functions. I can see a non-essential information functions being a security risk that could bring the whole system down… (Rob Healy)

I think VAM is very thorough. Also, way too complicated I believe that we just need to break the VAM into four core different essential steps Identify, Analyze, Mitigate, and Manage. Each step would involve the needed substeps involved with our Vulnerability assessment model. (Alexander Cannell)

I have to agree with Alex. Let’s simplify it so it can be a much speedier process. (Dion Munk)

Taking a little from both John, Rob, and Alexander’s comments, I’d offer the following to start:

1. Identify
   1. Group and rank various information functions
   2. Form policies based on these functions and rankings
   3. Prioritize mitigation/prevention activities based on most essential functions first
2. Analyze
   1. Analyze activities/traffic and form baselines for ‘normal’
   2. Monitor for vulnerabilities, compromises, departures from ‘normal’
3. Mitigate
   1. Take measures to fix problems found
   2. Alter policies if needed
4. Manage
   1. Document all fixes/changes to systems
   2. Add/alter policies and groupings/rankings as business grows and changes

(Josh Foremaster)

Josh I like your simplified model. Would something like penetration testing fall under section II B? Also, should a risk assessment be included in the model to help toward section I C (prioritize mitigation efforts)? (Bryce Caine)

In my opinion, policy should be first. Policy sets the risk appetite for the organization. It’s not really up to the security grunts to decide what level of risk is acceptable. That’s a business decision and should be left for the C-Suite. Once management defines the level of risk, then we can begin to implement controls that meet that risk. Risk management is really risk reduction, not risk elimination, so we need to know what level of risk we are shooting for. I also agree with Bryce. Some kind of risk assessment should be included. That might be part of the Identify section. You really should start with those systems that have the greatest risk. Maybe that’s implied when we determine the “most essential functions.” (Mark Walton)

The C-Suite might give the stamp of approval for acceptable risk, but I don’t feel C-level administrators know enough of the details of the data and systems that their organization uses to understand what is high risk data and what is not. I think the security team needs to assess risk and present their recommendations to the top level administration. They can then make an informed decision about risk and what they are willing to tolerate. So, that would imply that identifying essential information functions does come before policy making. The policy needs to be informed by something. I actually like Josh’s proposed model. The only thing I would add would be to repeat the process. (Parker Grimes)

I think the baseline is essential to any VAM. You have to know what normal is and what your tolerances will be. Those tolerances may change over time, but initially I would think they would be pretty tight. I think after each incident/test, the tolerances should be re-evaluated (as part of IV B) to make sure they are still accurate. (Jared Hallows)

I have to agree with Mark. Policy needs to be first. A good policy to start with will ensure things get done right from the get-go. I like Josh’s simplified model. I like leaning in the direction of the Operational Risk Management model a little bit on some of the VAM elements. Josh’s model is similar to the simple model Alex posted in our discussion last week, which I think is great. But, moving policies to the front, then ranking systems based on policy, and then reviewing the policy later on to make sure it is what it needs to be, I think, is the way to go. (Dion Munk)

I’m on the fence about whether policy or identification should go first. I can agree that policy should dictate what are the most valuable information and/or assets to protect. In a start up company however it might be more important to identify what will need protection. (Shea Esplin)

I’m on board for simplifying it. Josh, I like your start to the model. I’d like to see Bryce’s suggestions added in there. II B. Penetration testing for vulnerabilities? Move what is currently B to C?

I D. Perform risk assessment on policies and functions? (John)

The Identification step might also include the people involved as well. Where we inventory all IT systems and their access to the network, we should account for all users and their access to those systems. Is this is implied in information functions?

We could add something about testing to either II or III. It might fit best in III as an effort to make sure mitigation steps will be successful. This is meant to be more of the broad test rather than the penetration tests suggested earlier. ( Mark Whittaker )

(Brad Carroll) I believe there are at least two ways to consider this model: 1) strategic value and 2) tactical implementation. The value of the VAM model should provide the most protection to highest value assets. The would require the Identification stage conduct a “value analysis” as part of the A. Group and Rank the Information functions, which may ultimately lead to the B. Policies statement. For example, if the organization has customer credit card data, the value of the CC numbers would rank much higher on the list than PII, which is valuable, but not an immediate threat. Because the value of assets would change over time, the prioritization will also change. Therefore, I think the loop should return back to Identify and be tied to budgeting cycles.

Also, I would suggest calling the last stage “monitor” in place of manage because the whole process is management. Monitoring implies certain activities and accountability.

No, the manage stage isn’t the monitoring stage, the identify is like the monitoring where it's waiting for something to be identified. The manage stage is where you would want to set your baselines, correlate between the baseline and this current attack, set a new baseline and install a preventative measure so it won't happen again hopefully. (Scott H)



I like the diagram, but I would add to the Manage a substep indicating the need of determining if fixes made were successful or not. which enables us to go back to the Analyze and Identify Steps from the Manage Step.Or do you think something needs to be placed between mitigate and manage? (Alexander Cannell)

I too like the diagram. I would add if information obtained by adversaires (mitigate) can/could be interpreted to be useful to them. OPSEC stuff. (Steve Jaques)

So what if we take what Josh included and build out our final draft? Once we get this down we can modify the diagram. (JL) Diagram edited.

That Diagram looks great now (Scott)

1. Identify
   1. Identify, group, and rank various information functions
   2. Form policies based on these functions and rankings
   3. Perform risk assessment
   4. Prioritize mitigation/prevention activities based on most essential functions first
2. Analyze
   1. Analyze activities/traffic and form baselines for ‘normal’
   2. Perform penetration testing
   3. Monitor for vulnerabilities, compromises, departures from ‘normal’
3. Mitigate
   1. Take measures to fix problems found
   2. Alter policies if needed
4. Manage
   1. Document all fixes/changes to systems
   2. Add/alter policies and groupings/rankings as business grows and changes
   3. Asses implemented fixes and monitor to ensure their success
   4. Repeat process