Marimba-Automated Patching

Table of Contents

1.0 Business Case

As the companies struggle with budget pressures in a tight economy, the importance of automated patching remains a prominent consideration in the allocation of IT budgets. Despite of the burden of allocating the budget rightly, Automated Patching must not be ignored or allowed to fall by the wayside as keeping patches up to date can protect companies from exposure to malicious malware.

Rather than relying on industry best practice recommendations for manually keeping all OS and applications up to date with patches, enterprise patch management software enables IT organizations to delegate that task to sophisticated software that automates patching and ensures that all computers remain up-to-date with the latest patch releases.

Automated Patching solutions can fit nicely into your vulnerability management program, drastically reducing costs and narrowing the window in which your mission-critical systems are exposed. The enterprises can benefit greatly from automation.

2.0 Scope of Automation Patching

Having an automated patch management solution in place can greatly reduce the risk of security threats to all the IT infrastructure assets. It gives you certainty that the security of your systems is top notch, immune to known vulnerabilities as well as in regulation with government compliance laws. Patching closes the risk of security breach caused due to exposure and therefore ensures better productivity and efficiency of the business.

3.0 Why Marimba?

Marimba can be used as a cost effective alternative to the drain of manual patching. Marimba enhances the endpoint security as automating patch management instantly and uniformly patches all the systems in the IT infrastructure thus reducing the risk of threats. There is no match to ROI that comes through automation of patch rollouts using a mature and reliable framework like Marimba. Since, it is an automated process, it eliminates the scope of any manual errors.

Marimba automated patching simplifies a ton of steps involved in the enterprise patch management process successively saving lot of time and effort. In addition to that it also increases productivity as it frees up technical resources for other mission-critical IT tasks.

Marimba is a comprehensive enterprise patch management solution that automates patching of all the computers in a given IT infrastructure and keeps the vulnerabilities at bay.

4.0 Factors effecting Automation Patching

There are multiple factors on which Automated Patching (or even patching) depends. These factors directly impact the decision of an organization to have a process of automated patching or not. These factors may be,

- The value of your protected assets
- The threat level

- The presence of other mitigating factors
- The required effort and resources

5.0 Advantages of Automated Patching

For most of the companies, the decision about using automated patching arises from measuring the cost of patching and not patching against the level of risk, and then determining when and what to patch.

Over period of time, analysts have identified few major reasons why any organization must automate patch deployment using product like Marimba: -

1. **Higher ROI**

   Analysts like Gartner and Forrester have spelled out an equation to compute the cost of Patching:

   Cost of one time patching = (Hours x Rate x Systems) + (Patch Failure% x (Hours x Rate x Systems))

   Let us now take an example of an Infrastructure with 5000 systems (very similar to Amadeus). If it takes an army of $70/hour technicians one hour to patch a system, the cost of one time patching is $350,000. Both Gartner & Forrester have found that about 5 percent of the patches fail in a typical environment. And an average of two hours of recovery time (which includes help desk and IT support activities) are generally incurred to fix them. In the above case, we are looking at 250 systems with failures. And at the rate of $140 each, we are looking at another $35,000.

   So the total cost of one time manual patching becomes **$385,000**

   Also, there will be recurring cost of at least 10% (even with basic scripting) to roll out new patches. That would mean additional cost of $38,500 every time new patches need to go out.

   Automated tools like Marimba reduces the 5,000 man-hours to a few automated hours. This gives an extremely higher ROI which can be impossibly ignored.

2. **Complexity of Environment**

   Even if we take out the commercial impact, there is a big reason why organizations need to rethink their patching strategy. It comes through some important factors:

   - Heterogeneity of environment
   - Distributed environment
   - Variations in platforms (including network) and
   - Configurations and deployed applications

   The failure chances (of patching) increases drastically. Heterogeneous environment would require domain knowledge. A distributed environment would need much longer to patch as it involves coordination, scheduling and synchronization. Platforms and network too contribute to delays and failure of patching. And finally, the system configuration and applications (settings and configurations) are vast and it needs knowledge about patches and combinations to make them work properly.

3. **When to patch**
   The decision about when to patch is extremely important, but often overlooked. You should patch at the earliest point in time where the cost to patch is less than or equal to the cost not to patch. For example, the risk to unpatched systems -- and the potential cost in downtime and recovery -- increases drastically once an exploit is publicly available. Manual patching would need dedicated resources to look out for patches notification by different application vendors and then use scripts to roll them out in the production environment. That is time taking process and is prone to mistakes.

4. **Consistent deployments and Rollbacks**
   It is also necessary to look at patches applicability too. Patches are tied to specific OS versions, application versions etc. They are superseded by another patches and it is important to install patches in right order. And it is equally important to roll back when there are problems.

   Organizations need to assess the risk (and cost) of not doing anything (or doing things partially). They should weigh the risk of leaving some systems unpatched. This is a far more complex determination, incorporating the cost of recovery for stricken systems and the value of the system at risk adjusted for the likelihood that it will be compromised. The value of the system, along with its corresponding risk of loss, is an element that continues to elude security professionals. Regardless, it's important to at least "gut check" the potential for loss.

   There are six key elements of loss for any computing asset:

   - Lost productivity for the end user
   - Lost productivity for IT support personnel
   - Loss of revenue (direct)
   - Legal/regulatory costs
   - Intellectual property losses
   - Loss of stored assets (financial)

   Each of these elements help you quantify the potential loss of an asset, whether the breach compromises confidentiality, data integrity/availability or system availability. With worms and viruses on desktops, for example, the losses are typically in the productivity of the desktop itself, but are exacerbated if the desktop infects the rest of the network. Next, consider the factors that determine the risk of an exploited vulnerability on the computing environment:

   - The impact of the vulnerability is about damage or the payload of the attack. Worms and viruses can delete files, install Trojan software, provide a root shell, and participate in a distributed DDoS attack.
   - Likelihood corresponds to the exposure of a system to exploit code.
   - Criticality is the functional value of the system to an organization.

6.0 Summary

It becomes important that organizations look up to automation of patch deployments (in a continuous mode). They should also look up for tools that offer updated definitions and patches together. Marimba patching saves time and money. Marimba consolidates metadata that contains newly discovered application and OS patches. They are released as periodic updates to customer. Customers can schedule auto update of these definitions. Once the definitions are updated, the machines get new definitions as per the schedule. Marimba customers have seen over 99% patching compliance within 48 hours of machine's availability. This takes care of infrastructure configurations and is extremely bandwidth friendly.