HIGH RISK

MEDIUM RISK

LOW RISK

THREAT-O-METER

PUBLIC CLOUD

LAPTOP WITH
BROWSER

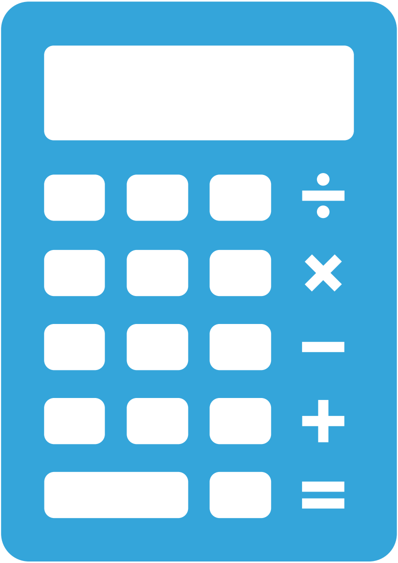
Exploit unlikely or running untrusted code already worst case

Exploit possible but needs another successful attack to run attackers code

Exploit possible and runs untrusted code “by design”

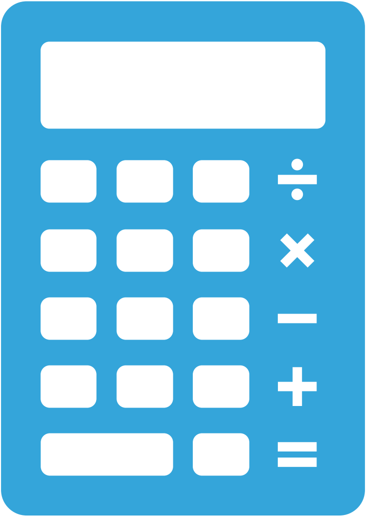
PRIVATE CLOUD

DATABASE SERVER

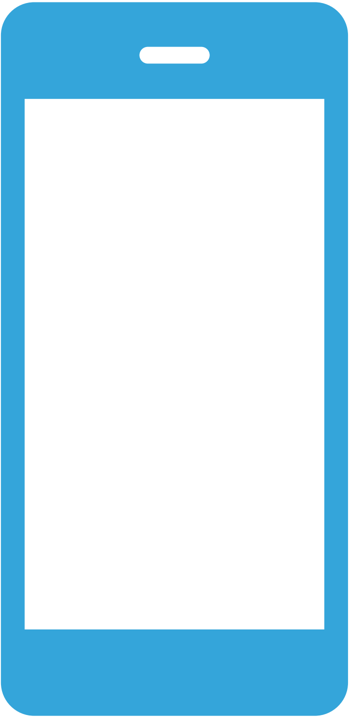


MAILSERVER

FIREWALL



APPLICATION SERVER



MOBILE PHONE

Public clouds run code of many untrusted parties which makes them very vulnerable.

Databases are often protected from the internet and are accessed only by application servers.

Running untrusted code on a database is often already the worst case scenario. Patching against Meltdown/Spectre would only marginally increase security.

Mailserver are exposed to the internet but have been proven to be very robust to “remote code execution” attacks.

Also a code execution is already the worst case.

Arguably mail servers can be placed in “medium” due to their exposure to the internet.

Laptops/desktop systems with browsers are very vulnerable because they execute untrusted code in the form of JavaScript from websites.

Threat - O - Meter

Mobile phones run apps and websites (JavaScript).

Firewalls and switches (normally) do not expose an attackable surface to the external network.

This greatly reduces the likelihood of attacks.

A code execution is already the worst case.

VPN gateways expose a complex interface and are more likely to be attacked.

Private clouds run many different workloads but they are all trusted.

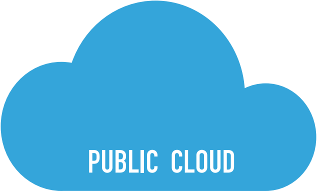
An attacker only needs to hack one application running in the cloud to run a Spectre attack.

Given the patches are risky w. regards to performance and availability.

What would be your patching strategy for each risk class?

Application servers only run trusted code but attacks can lead to code execution.

How many Java (node, Ruby,..) **libraries does your software use? And transitively? Who audits all these?**

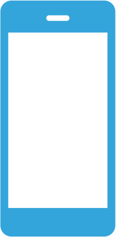


PUBLIC CLOUD

DATABASE
SERVER

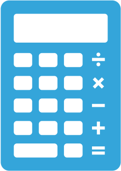
MAILSERVER

LAPTOP WITH BROWSER



MOBILE PHONE

FIREWALL



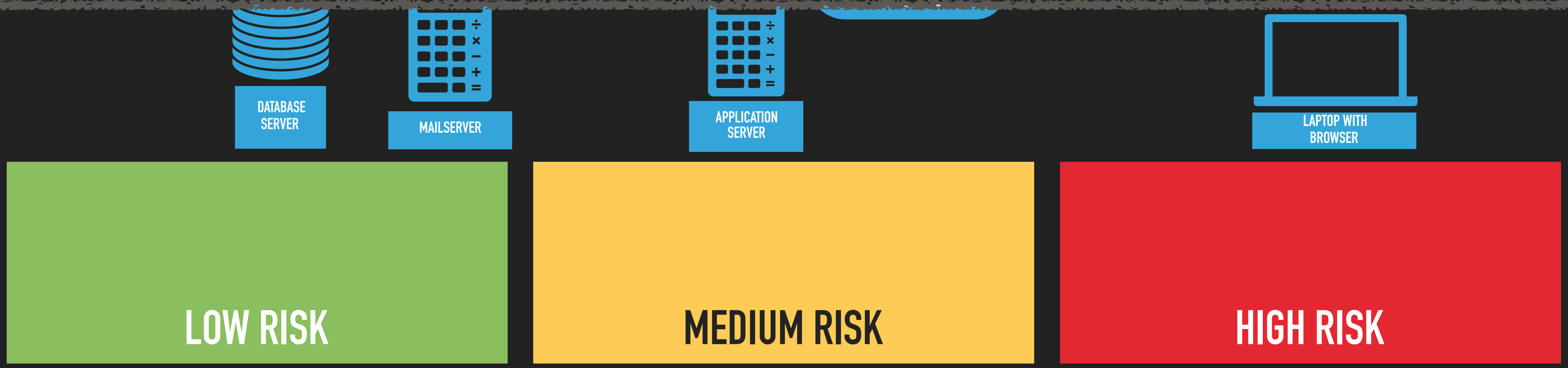
APPLICATION
SERVER

PRIVATE CLOUD

THREAT-O-METER

Given the patches are risky w. regards to performance and availability.

What would be your patching strategy for each risk class?



DATABASE SERVER

MAILSERVER

APPLICATION SERVER

LAPTOP WITH BROWSER

LOW RISK

MEDIUM RISK

HIGH RISK

Exploit unlikely or running untrusted code already worst case

Exploit possible but needs another successful attack to run attackers code

Exploit possible and runs untrusted code "by design"

ACCIDENT, MALICE, INCOMPETENCE?

WHY DID IT HAPPEN?