2. (2 points)

Company Safety Inc develops software for different kinds of power plants. They are specialized in the software that controls power plants' safety-critical operations.

Describe how they could use security architecture for software development.

Describe those items of security architecture that you consider especially important. Pick at least 4 items.

Aim for highest security level

Set the high-level design principles that guide the software development process, from design, implementation, testing to maintenance.

Design before implementation

Building from functional requirements and high level design principles create an overall architecture. Keeping the system running safely and continuously is the most important goal.

Design enemy in mind

Create attack scenarios: finding attack vectors that are not yet covered. Be sure to gather the latest reports on existing attacks and verify (or create) current protections against those attacks. (Getting information can be difficult due to the security nature of the industry.)

Apply defense in depth

Create a multilayer security model that takes into account the influence of non-software (or hardware) related incidents, e.g.:

- Human planned or accidental misuse of the system
- False alarm caused incorrect reporting of the system components

System must be able to detect and recover from these kinds of events.

Well-defined access rights and sharing responsibilities

An essential part of the system is how it will be used and by who. Access rights must be clearly defined on every level and for all operations. An essential part of this definition work is to create a set of user roles, but keeping in mind the principle "Aim for simplicity".

Fault tolerance

Never rely on a single resource or component, instead every component must have multiple backup resources.

Alternative resources

There is no such thing as "too much redundancy". Backup resources are needed on every level of the system, both software and hardware; e.g. software implementations on separate servers and different OS's.

Manage errors, recovery

Create detailed and well-communicated action plans describing what measures must happen to recover from events, either automated or manual. See Trace events.

Risk management

Major risks (caused by the software system) were identified already in the early design stages; those plans should cover on high level mitigation plans. On more detailed levels potential risks should be discovered and avoided, and the residual risks should be clearly communicates.

A [nuclear] powerplant software operates closely with hardware components and the supply chains of these components must be considered as a potential source of defects: changes in their APIs or other configuration settings can cause catastrophic situations.

Trace events

Event logging (along with real time monitoring) is a necessary tool for resolving problem situations and also monitoring normal operating status.

Sources

Lecture materials

https://www.sciencedirect.com/science/article/pii/S0149197023001737