**Metodologie**

Úloha č. 1

Vypracoval: Vávra Tomáš

**Co je to kyberdoména/kyberprostor:** Na tuto otázku lze nahlédnout hned z několika pohledů, které se ani samotné úplně neshodují. Jako první definoval kyberprostor Willian Gibson (Gibsonův kyberprostor) v díle Neuromancer, avšak sám autor svou definici později kritizuje až jako nesmyslnou. V dnešní době nalezne asi největší uplatnění výklad zákona, který o kyberprostoru hovoří jako o digitálním prostředí skládajícím se z informačních systémů, služeb a sítí elektrotechnických komunikací umožňujících vznik, zpracovaní nebo výměnu dat. Osobně jsem tento pojem nejlépe pochopil jako abstraktní/fyzicky nereálný prostor, do nějž nejrůznější aktéři vstupují za účelem práce s daty (vytváření, upravování, sdílení, selektování, atd.) a to za pomoci nejrůznějších moderních technologií umožňujících vstup do tohoto prostoru.

**Definování problému v kyberprostoru, kde lze využít M&S:** Jako problematiku tohoto úkolu jsem si zvolil M&S testování SW zabezpečení fyzických cílů s příliš vysokým významem pro volné bezpečnostní testovaní. Tento problém spočívá především v požadavku na ideálně nepřetržitou funkčnost/připravenost určitých systémů (zbraňové systémy, radary, apod.), kde by určitá bezpečnostní testování SW mohla způsobit nepředvídatelné disfunkce systému.

**V čem by M&S spočívalo:** Jednalo by se o vytvoření testovacího prostředí s modelem daného systému (SW i HW náležitosti), který by umožňoval volná bezpečnostní testování, vůči jakýmkoliv hrozbám z kyberdomény nebo předvídání nežádoucích efektů SW údržby. Nasimulování nejen SW, ale i HW by mělo mít efekt částečného odstranění „slepých úhlů“ pro testery, kteří by tak mohly předcházet potenciálním problémům vznikajících při virtuálním testování SW, kde můžeme snadno přehlédnout určité dopady, které mají neočekávané situace nastávající na úrovní SW na HW.

**Pokus o odhad daného řešení:** Toto řešení problému by bylo velmi závislé na přesnosti a kvalitě duplikace systému, jeho náležitostí a případných změn vyplívajících z potencionálních změn samotného systému, a tak by tento způsob řešení mohl být do určité míry sebedestruktivní vlivem své vlastní funkce. Představa ideálního fungování tohoto řešení by byla taková, aby jakékoliv předměty testování, vyhodnocování, údržby, apod. byly nejdříve otestovány v testovacím prostředí a následná řešení vyladěny vůbec před stykem s „živým systémem“, čímž by se docílilo minimalizace doby plánované nefunkčnosti/omezené funkčnosti systému. Výsledkem by tak nebyl „jen“ spolehlivější a bezpečnější systém, ale i zmenšení nákladů potřebným pro jeho odstávky (otázkou je zda-by je provoz testovacího prostředí nepřevýšil)