Attack Trees

The security needs of any valuable information system or asset are not static. To maintain continuous security of the information assets and information systems, the defender needs to defend against growing attacks continuously. A fault tree analysis was developed earlier to evaluate systems' safety, followed by similar structures to enhance systems security engineering. Attack trees were one of the structures; the architectural design of the attack tree consisted of the root of the tree and child nodes represented as leaves(Mauw & Oostdijk, 2006). The root of the modelling represents the security threat, modelled as the attacker's primary goal. Logic gates are used to refine the root into attackers subgoals, and these models show how the successful phases propagate through the system (Fraile et al., 2016). When a particular step requires complete success in all of its nodes by the attacker to succeed, it is modelled by the AND-gates; when an action requires the attacker to achieve in one or more on the child nodes, it is modelled by the OR-gates.

Attack-defence trees

The attack trees limit the precision with which adequate defensive strategies can be analysed since this model does not account for the efficacy of existing countermeasures that may mitigate new attacks. Attack-defence trees overcome those limitations by introducing defensive countermeasures and actions against attacks (Abbas et al., 2021). Countermeasures are modelled as a child opposite type of the nodes. The key features of the AD-Tree are refinements and countermeasure. Refinements represent a subgoal of the attacker and are modelled as either disjunctive or conjunctive. Analysing the attack-defence scenario is the purpose of the AD-Tree (Beaulaton & Beaulaton, 2020).

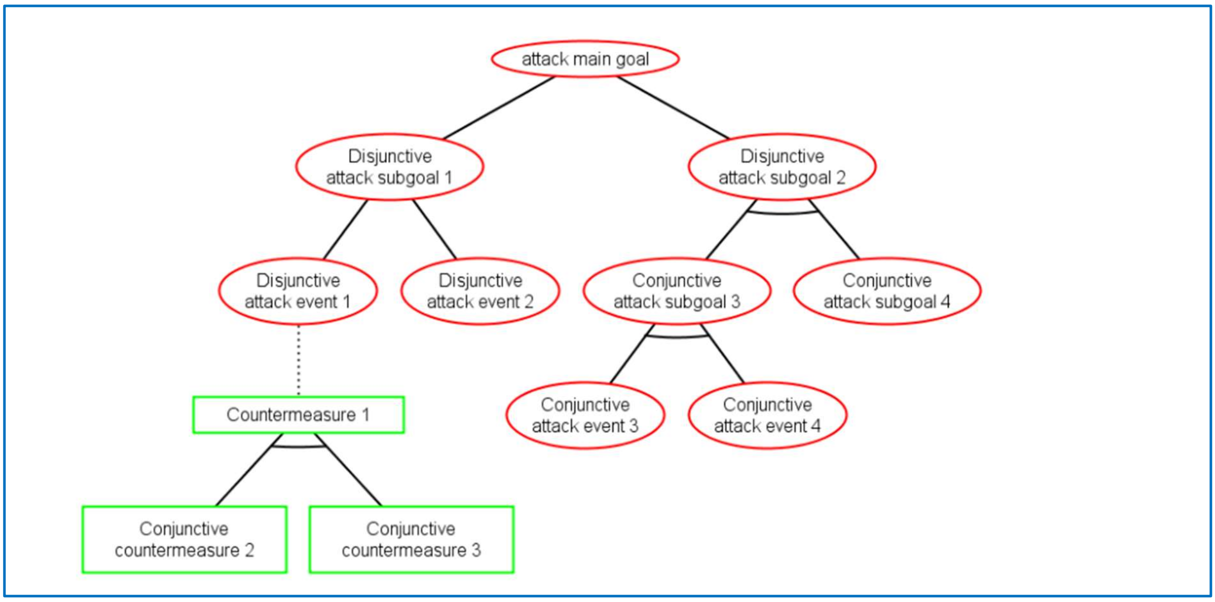
attack main goal
Disjunctive attack subgoal 1
Disjunctive attack subgoal 2
Disjunctive attack event 1
Disjunctive attack event 2
Conjunctive attack subgoal 3
Conjunctive attack subgoal 4
Countermeasure 1
Conjunctive attack event 3
Conjunctive attack event 4
Conjunctive countermeasure 2
Conjunctive countermeasure 3

Figure 1: General Structure of an attack defence tree (ADT)

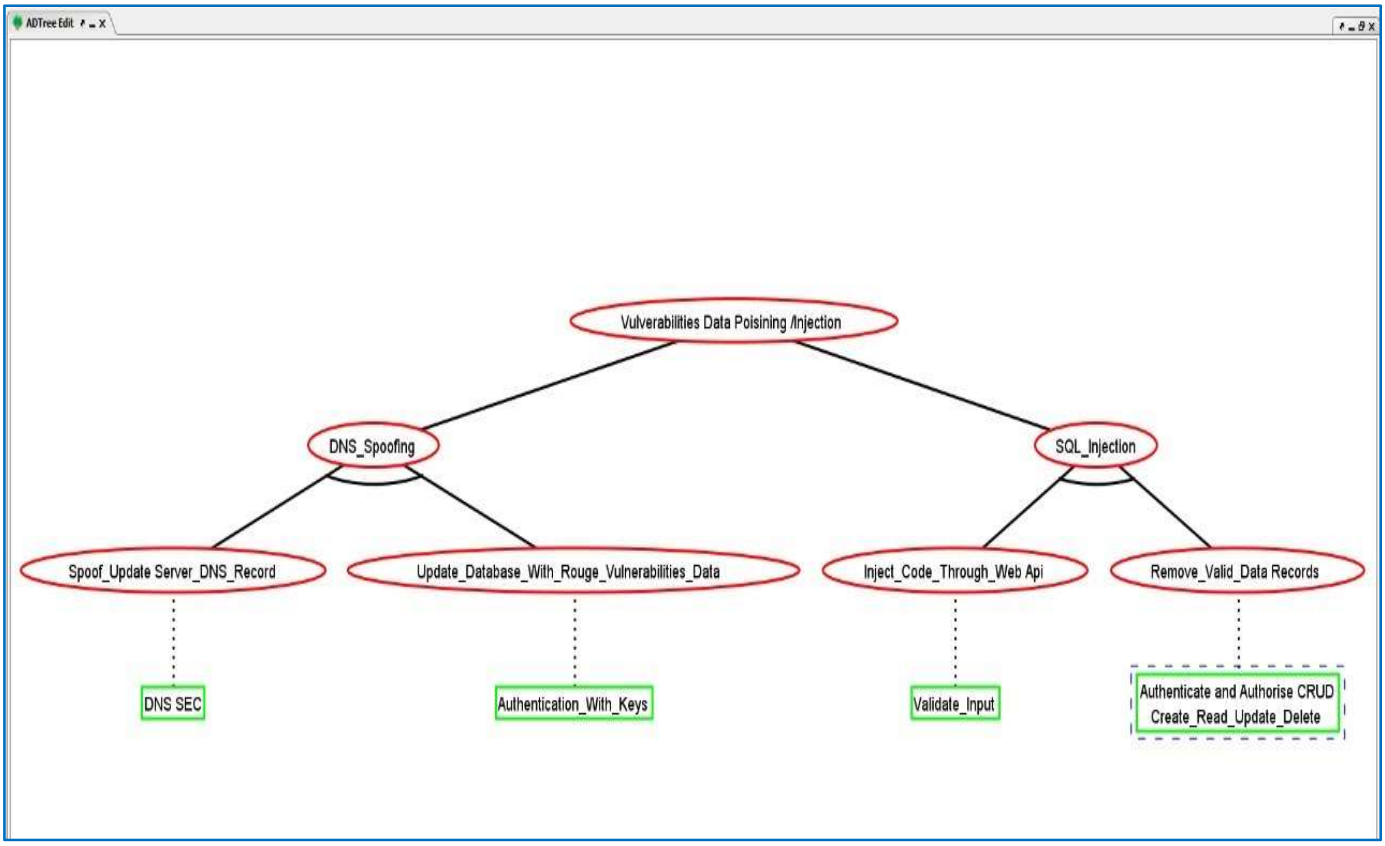
ADTree Edit
Vulverabilities Data Poisining /Injection
DNS_Spoofing
SQL_Injection
Spoof_Update Server_DNS_Record
Update_Database_With_Rouge_Vulnerabilities_Data
Inject_Code_Through_Web Api
Remove_Valid_Data Records
DNS SEC
Authentication_With_Keys
Validate_Input
Authenticate and Authorise CRUD
Create_Read_Update_Delete

Figure 2:Structure of Data/Record poisoning (ADT)

Domain Name Server (DNS) spoofing

Domain Name Server (DNS) spoofing (a.k.a. DNS cache poisoning) is an attack in which altered DNS records are used to redirect online traffic to a fraudulent website that resembles its intended destination.

Once there, users are prompted to login into (what they believe to be) their account, allowing the perpetrator to steal their access credentials and other types of sensitive information. Furthermore, the malicious website is often used to install worms or viruses on a user's computer, giving the perpetrator long-term access to it and the data it stores.

Methods for executing a DNS spoofing attack include:

- Man in the middle (MITM) – The interception of communications between users and a DNS server to route users to a different/malicious IP address.
- DNS server compromise – The direct hijacking of a DNS server, which is configured to return a malicious IP address.

DNS spoofing mitigation using domain name server security (DNSSEC)

DNS is an unencrypted protocol, making it easy to intercept traffic with spoofing. What's more, DNS servers do not validate the IP addresses to which they are redirecting traffic.

DNSSEC is a protocol designed to secure your DNS by adding additional verification methods. The protocol creates a unique cryptographic signature stored alongside your other DNS records, e.g., A record and CNAME. Your DNS resolver then uses this signature to authenticate a DNS response, ensuring that the record wasn't tampered with.

Bibliography:

Abbas, S. G., Vaccari, I., Hussain, F., Zahid, S., Fayyaz, U. U., Shah, G. A., Bakhshi, T., & Cambiaso, E. (2021). Identifying and Mitigating Phishing Attack Threats in IoT Use Cases Using a Threat Modelling Approach. *Sensors*, *21*(14), 4816. https://doi.org/10.3390/s21144816

Beaulaton, D., & Beaulaton, D. (2020). *Security Analysis of IoT Systems using*

Attack Trees To cite this version : HAL Id : tel-02893847.

Fraile, M., Ford, M., Gadyatskaya, O., Kumar, R., Stoelinga, M., & Trujillo-Rasua, R. (2016). Using attack-defense trees to analyze threats and countermeasures in an ATM: A case study. *Lecture Notes in Business Information Processing*, *267*, 326–334. https://doi.org/10.1007/978-3-319-48393-1_24

Mauw, S., & Oostdijk, M. (2006). Foundations of attack trees. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *3935 LNCS*(September), 186–198. https://doi.org/10.1007/11734727_17