22.2.24 ইন্টারভিউ ও সাইবার সিকিউরিটি

অনুষ্টুপ@gmail.com - 1120 ১৮৩২

সাইবার সিকিউরিটি কেন গুরুত্বপূর্ণ?

ডিজিটাল সমাজে তথ্য হচ্ছে সবচেয়ে মূল্যবান সম্পদ। ব্যক্তিগত তথ্য, ব্যবসায়িক তথ্য, সরকারি তথ্য সবকিছুই সাইবার অপরাধীদের লক্ষ্য। তথ্য চুরি, ডেটা ফাঁস, ম্যালওয়্যার আক্রমণ, ransomware — এগুলো প্রতিনিয়ত বাড়ছে। সাইবার সিকিউরিটি ছাড়া আধুনিক বিশ্ব অচল।

CIA Triad (সিআইএ ট্রায়াড)

সিকিউরিটির তিনটি মূল স্তম্ভ হলো Confidentiality, Integrity, Availability। Confidentiality মানে তথ্য শুধু অনুমোদিত ব্যক্তি দেখবে। Integrity মানে তথ্য পরিবর্তন করা যাবে না। Availability মানে সিস্টেম সবসময় কাজ করবে।

Malware (ম্যালওয়্যার)

Malware হলো ক্ষতিকর সফটওয়্যার। Virus, Worm, Trojan, Spyware, Ransomware — সবগুলো malware। Ransomware তথ্য লক করে দেয় এবং মুক্তিপণ দাবি করে। Trojan নিজেকে বৈধ সফটওয়্যার হিসেবে লুকায়।

Phishing (ফিশিং)

Phishing হলো ভুয়া ইমেইল বা ওয়েবসাইট দিয়ে ব্যবহারকারীর তথ্য চুরি করা। ব্যাংক, সরকার, বা বিশ্বস্ত প্রতিষ্ঠানের নকল সেজে ব্যবহারকারীকে ফাঁদে ফেলা হয়। Spelling mistake, suspicious link, urgent message — এগুলো phishing এর লক্ষণ।

Firewall (ফায়ারওয়াল)

Firewall হলো network traffic control করার system। Firewall নির্ধারণ করে কোন data ভেতরে ঢুকবে আর কোন data বাইরে যাবে। Firewall ছাড়া network খোলা দরজার মতো।

Encryption (এনক্রিপশন)

Encryption হলো data কে এমন code-এ পরিবর্তন করা যে শুধু সঠিক key দিয়ে তা পড়া যায়। Key ছাড়া encrypted data অর্থহীন। Encryption ছাড়া online banking, messaging, e-commerce সবকিছু ঝুঁকিতে পড়ত।

Firewall ও Antivirus এর পার্থক্য

Firewall network level-এ কাজ করে। Antivirus device level-এ malware ধরে। Firewall বাইরের আক্রমণ আটকায়, Antivirus ভেতরের ক্ষতি দূর করে। দুটোই আলাদা কাজ করে।

ב open source הכל פתוח וכולם יכולים להגיע לקוד. זה גם כי
יכולים יותר להגן אבל זה גם פרוץ יותר.

current->uid=0 מנסה להגיע ל root. יש רק = אחד, זה עושה
השמה ואם זה שווה true לקוד. המשתמש שנכנס עכשיו, אז הופך להיות
הוא יכול עכשיו להגיע להרשאות הכי גבוהה.
מנסים לעשות back door?
מה זה back door?

רוצים להשיג את מידע פרטי וקבצים, פרטי חשבון בנק, סיסמאות, מידע
אישי שיכול להגיע לגניבת זהות, מיקום.

Phishing
בניית אתר שנראה כמו אתר מוכר
מה זה phishing?

CIA- confidentiality, integrity and availability
מידע חסוי, שלמות, זמינות
מידע חסוי- רוצים שהמידע יהיה מוגן (הצפנה), וכו'.
שלמות- מה שהלקוח שלח יגיע ליעד.
זמינות- השרת יהיה תמיד זמין.
authentication - זיהוי. מאמת את האדם שזה אכן הוא. סיסמא, טביעת
אצבע. חייב זהות פנים כיווני.
non repudiation - אי התכחשות.

הצפנה- לוקחים data ומערבבים בעזרת מתמטיקה, אחר כך מה input:
זה מפתח ש-2 האנשים יודעים אותו. המידע יכול להגיע לאדם
עם sniffers שיכול להקשיב למה שמדברים, אבל זה מוצפן וצריך
את המפתח.
אם מרגישים סימן שאלה עשו זיהוי טוב.
Protection = prevention + (detection + response).
תמיד צריך לשאול את עצמי מי יעזר לי

prevention - המנעות. מניעת התקפות.

detection - מודעות שמישהו תוקף אותי.

response - הגבה. איך אני מגיב לתקיפה. למנוע תוך כדי שתוקפים כי אפשר לזהות.

כפי שרואים מיגון צריך לעשות את כל ה-3.

יש גבול להשקעה, לא נשקיע יותר ממה שעולה.

תמיד מנסים המנעות.

1. prevention - הצפנת מידע עוזר למניעת גניבת מידע.

firewall - חומת אש. צריך להגדיר כמו שצריך. מגן תוכנתי שמשמש שיושב בכניסה לרשת ומסתכל על כל החבילות שנכנסות ומחליט אם החבילה תיכנס או לא. רואה מה יש בחבילה ומחליט אם תיכנס. מסתכל על חבילות יוצאות ונכנסות.

access control - מתן הרשאות. הגיע אדם ועושים לו authentication. מאפשרים לעשות דברים מסוימים, חוקים מה מותר ומה אסור.

2. detection - IDS. מערכת שיודעת לזהות חדירות לרשת. משהו הדעיה שחבור ורוצים לדעת עליו. כל חדירה, תקיפה כעת מתריעים וצריך לזהות.

honey pots - מלכודת דבש. מכינים מטרה מפתה, משאירים קצת פתוח מיגון כדי שיפרצו ויגלו את הפורץ.

3. response - תוקפים אותי, זיהיתי מי זה ועכשיו מגיבים, תגובה מיידית. עוצרים שרשרת התקיפה (גם אם עוד לא יודעים מי) ומפסיקים (מנתק חשמל). כשיודעים מי פורץ חוסמים אותו. שינויים בתוכנות לא טובים. לא יודעים מה קרה ומה הפורץ ניסה לעשות, עוקבים אחרי הפורץ לראות מה הוא עושה לאחר ומנסים להגיע למי שמפעיל ולמנוע להתארגן מולו. זיהיתי שהצפינו את המידע.

normal flow - המידע עובר רגיל.

interruption - המידע לא מגיע לצד השני. פוגע ב-availability.

interception - האזנה, אי אפשר לשנות. פוגע ב-confidentiality.

modification - המידע עובר בדרך, אפשר לשנות את המידע. פוגע ב-integrity.

יש את שניהם. man in the middle

fabrication - להמציא משהו בשם מישהו אחר. פוגע ב-authentication.

ממציאים משהו, מתחזים.

איך שומרים על confidentiality - הצפנה, הגנה פיזית, הגנה בסיסמא.
איך שומרים על availability - אנטי וירוס, שמירות גיבויים, ביטוח
איך שומרים על non repudiation - חתימה דיגיטלית
איך שומרים על access control - סיווגים classification, מדרגים אותם ובודקים איזה הרשאות יש.

MOM: 3 דברים לתקיפה מוצלחת
motive - איזה סיבה יש לי לתקוף משהו. לפעמים סתם רוצים לראות שאני יכול.
opportunity - הזדמנות. בפועל שיש לו הזדמנות, לא משנה אם זה חשוב
method - השיטה שבה אני הולך לתקוף, לפרוץ

3 גישות לבעיית אבטחת מידע:
1. ignore - למה שמישהו יתקוף אותי, שעדיף שיש משהו אז נגנב משהו. out of the box. לא עושים שום דבר.
2. host security - מגינים על כל אחד ואחד. מגינים בצורה שונה על אחרים. הגנות שונות על שרתים שונים.
3. network security - מגינים על כולם ביחד. למשל, שמים firewall. לא מגינים על כל אחד בנפרד.
מה יותר טוב? host הכי טוב כי מתאימים לכל אחד לפי הצרכים שלהם. אבל host זה יקר, וקשה להשתלט על זה כשיש הרבה. network יותר חלש אבל יותר פשוט.
נשים גם וגם. נגן בצורה הכי טובה ברמת ה network ומי שחשוב נשים לו host. זה מקשה על התוקף, זה יותר שכבות. זה יכול לעכב את הפורץ או למנוע ממנו.
אם כל ה firewall משכבות אותו דבר ומצאו פגיעה אפשר לעבור הכל.
diversity of defence - נשים vendors שונים (למשל firewall), גיוון של סוגי הגנות. כשזה vendors שונים אם פוגעים ב1 אי אפשר לפרוץ את האחר כי הפגיעות שונה.

how a layered approach would work
intruder - פורץ.

penetrate - לחדור

תוקף צריך לראות איך הוא יכול לחדור לחומת אש. לאסוף מידע על החברה וחומת האש. לאחר החדירה, ההתנהגות צריכה להיות נורמלית שלא ידעו שהוא שם". צריך לעבור כאילו שמישהו סינן החבילות שנחשב גם חומת אש. כשאני בפנים ומנסים לעשות authentication מול DB אפשר לעשות brute force attack וכך לנסות את כל הסיסמאות האפשריות. זה ארוך ויכולים לעלות עלי בסוף כי מנסים הרבה סיסמאות

social engineering - לקחת מידע מהאדם. אפשר לבנות מילון שמכיר לעצמם ואז לעשות brute force attack כי זה מצומצם אופציות

הצלחתי לעשות authentication והגעתי למשתמש, אבל אולי הוא משתמש פשוט ומערכת access control מגבילים אותו בגישה. אפשר לנסות לעלות הרשאות או לעבור.

כדי לעשות access control יש רשימות של הרשאות לכל אחד ואפשר לנסות לפרוץ אותה ולתת למי שצריך להגיע או לשנות הרשאות למשתמש.

principle of least privilege

מינימום הרשאות שצריך כדי לעבוד. חשוב מהמקום התקפה או שמישהו יפגע בעובד ויכנס הלאה.

רוב הפריצות הן פריצות מבפנים שנעשות בטעות.

שאלה: מה ההבדל בין access control ל-authentication? מה קודם?

קודם auth ואז access control,

auth - מזהים את האדם.

access control - רשימה של הרשאות.

מצגת 1 עמ' 25 תשובה

authentication

3 שיטות:

1. משהו שאני יודע - סיסמא

2. משהו שיש לי - כרטיס חכם, טוקן, טלפון (סמס או קוד).

3. משהו שאני - בצורה ביומטרית. טביעת אצבע, סריקת קרנית.

פרק פתוח 2.

* ভাঁজ ও মডুলাস ।

ভাঁজ সংজ্ঞা ও

ভাঁজ : কোনো সরল দণ্ড বা বিম যখন বাইরে থেকে প্রযুক্ত বলের প্রভাবে বেঁকে যায় তখন তাকে ভাঁজ বলে।

ভাঁজের ক্ষেত্রে বিমের উপর প্রযুক্ত বলের প্রকৃতি - বিমের উপর প্রযুক্ত বল হতে পারে স্থির বা চলমান।

ভাঁজের প্রকার :

ভাঁজ প্রধানত দুই প্রকার। যথা - স্থির ভাঁজ ও চলমান ভাঁজ।

স্থির ভাঁজ : যে ভাঁজ বিমের উপর স্থির বলের প্রভাবে সৃষ্টি হয় তাকে স্থির ভাঁজ বলে।

চলমান ভাঁজ : যে ভাঁজ বিমের উপর চলমান বলের প্রভাবে সৃষ্টি হয় তাকে চলমান ভাঁজ বলে।

স্থির ভাঁজের ক্ষেত্রে বিমের উপর প্রযুক্ত বলের প্রকৃতি - বিমের উপর প্রযুক্ত বল হতে পারে স্থির বা চলমান।

স্থির ভাঁজের প্রকার :

স্থির ভাঁজ প্রধানত দুই প্রকার। যথা - স্থির ভাঁজ ও চলমান ভাঁজ।

স্থির ভাঁজ : যে ভাঁজ বিমের উপর স্থির বলের প্রভাবে সৃষ্টি হয় তাকে স্থির ভাঁজ বলে।

চলমান ভাঁজ : যে ভাঁজ বিমের উপর চলমান বলের প্রভাবে সৃষ্টি হয় তাকে চলমান ভাঁজ বলে।

স্থির ভাঁজের ক্ষেত্রে বিমের উপর প্রযুক্ত বলের প্রকৃতি - বিমের উপর প্রযুক্ত বল হতে পারে স্থির বা চলমান।

স্থির ভাঁজের প্রকার :

স্থির ভাঁজ প্রধানত দুই প্রকার। যথা - স্থির ভাঁজ ও চলমান ভাঁজ।

স্থির ভাঁজ : যে ভাঁজ বিমের উপর স্থির বলের প্রভাবে সৃষ্টি হয় তাকে স্থির ভাঁজ বলে।

চলমান ভাঁজ : যে ভাঁজ বিমের উপর চলমান বলের প্রভাবে সৃষ্টি হয় তাকে চলমান ভাঁজ বলে।

15.2.24 ইন্ট্রোডাকশন টু সাইবার সেকিউরিটি

ইন্টারনেট অফ থিংস (IoT) হলো এমন একটি ব্যবস্থা যেখানে বিভিন্ন স্মার্ট ডিভাইস একে অপরের সাথে তথ্য আদান-প্রদান করে। ধরুন আপনার বাসায় একটি স্মার্ট থার্মোস্ট্যাট আছে যা আপনার মোবাইল ফোনের সাথে যুক্ত। আপনি অফিস থেকে বাসার তাপমাত্রা কমিয়ে দিতে পারেন। এই ডিভাইসগুলো ইন্টারনেট বা নেটওয়ার্কের মাধ্যমে একে অপরের সাথে কথা বলে। এর ফলে আমাদের জীবন অনেক সহজ হয়ে যায়। কিন্তু এই সংযোগের কারণে সাইবার অপরাধীরা এই ডিভাইসগুলোকে হ্যাক করে আমাদের ক্ষতি করার সুযোগ পায়। তাই আমাদের এই ডিভাইসগুলোর নিরাপত্তা নিশ্চিত করা খুবই জরুরি।

ডিভাইস ও নেটওয়ার্ক নিরাপত্তা

আমাদের বাসায় বা অফিসে যে ওয়াই-ফাই রাউটার ব্যবহার করা হয়, সেটি হলো নেটওয়ার্ক নিরাপত্তার প্রথম ধাপ। রাউটারের ডিফল্ট পাসওয়ার্ড পরিবর্তন করা, ফার্মওয়্যার আপডেট করা এবং গেস্ট নেটওয়ার্ক ব্যবহার করা খুবই গুরুত্বপূর্ণ। স্মার্ট ডিভাইসগুলোর ডিফল্ট পাসওয়ার্ড পরিবর্তন করা, অপ্রয়োজনীয় ফিচার বন্ধ রাখা এবং ডিভাইসের ফার্মওয়্যার নিয়মিত আপডেট করা নিরাপত্তার অন্যতম গুরুত্বপূর্ণ বিষয়।

ডেটা এনক্রিপশন এবং প্রাইভেসি

ডেটা এনক্রিপশন হলো তথ্যকে এমন কোডে রূপান্তর করা যা শুধু অনুমোদিত ব্যক্তিরা পড়তে পারে। ধরুন আপনি একটি চিঠি লিখলেন যা শুধু আপনার বন্ধু পড়তে পারবে। এনক্রিপশন ঠিক সেভাবেই কাজ করে। ডেটা ট্রান্সমিশন এবং স্টোরেজ উভয় ক্ষেত্রে এনক্রিপশন ব্যবহার করা হয়। প্রাইভেসি বলতে বোঝায় ব্যক্তিগত তথ্য কে সংগ্রহ করছে এবং কেন করছে তা নিয়ন্ত্রণ করার অধিকার।

ফিশিং এবং সোশ্যাল ইঞ্জিনিয়ারিং

ফিশিং হলো প্রতারণামূলক ইমেইল বা মেসেজ যা ব্যবহারকারীকে ভুলিয়ে ভালিয়ে ব্যক্তিগত তথ্য দিতে বাধ্য করে। সোশ্যাল ইঞ্জিনিয়ারিং হলো মানুষের আবেগ ও বিশ্বাস কাজে লাগিয়ে তথ্য হাতিয়ে নেওয়া। উদাহরণ হিসেবে বলা যায় কেউ ফোন করে নিজেকে ব্যাংক কর্মকর্তা পরিচয় দিয়ে আপনার পিন নম্বর চাইতে পারে। এই ধরনের আক্রমণ থেকে সাবধান থাকা খুবই জরুরি।

ধন্যবাদ!

কিন্তু 'ক্লাসিনো অফ দ্য হেভেন্স' নামের মতো হেভেনলি
না, মানে নেগেটিভ ইমেজ আসি
কিন্তু নতুন প্রজন্ম না জানলেও পুরনো দিনের যারা বেশি বয়স্ক আছি তো প্রেমে
মাটি পাপের ফল। কেউ
এমন কোনো মধ্যবয়স্ক-টয়স্ক নয়।
যেজন্য সাহেবের কাছে প্রস্তাব দেওয়া, যে, এটা একটা প্রকল্প হতে পারে
হাতে হাতেই মানুষ কে দিয়ে প্রজন্ম তৈরি করা যায়।
ক্লাসিনো কি আসলে একটা 'হয়'
প্রজন্ম পুরনো হলে, তাদের মধ্যে যারা টুকে যায়, তাদের মধ্যেও
তাদের ক্লাসিনো কি আসলে পাপের ফল
নতুন প্রজন্মের কি ক্লাসিনো
কাজ কি আর পাপের না
গাছের প্রতি তো
সাধন কি কাজের

25.2.18 intelligence gathering

עושים איסוף אינפורמציה בכל התקפה.
אוספים מידע מקדים, הכל מתחיל איתו כי יכול לעזור בהחלטה ובאיך לתקוף,
בסוג התקפה.

OSINT - open source intelligence:
כולל איתור, בחירה ואיסוף מידע ממקורות ציבוריים.
יש 3 אפשרויות לאיסוף אינפורמציה
passive - איסוף אינפורמציה ואף אחד לא יודע.
semi passive - חוקי ויכולים לגלות עליי.
active - סביר שיגלו עליי.
ואיסוף אינפורמציה מלמעלה למטה.
צריך לדעת איזה מטרות ומערכות הרשת יש לה. צריך לדעת מה אני רוצה
לעשות וזה ע"י לשאלה. רוצים להשיג כל מה שאני רוצה ושלא יגלו עליי.

שיעורי בית:
להוריד מערכת הפעלה בשם kali ולהתקין ב VM.
מורידים vmware (מכונה וירטואלית) עליה מתקינים מערכת הפעלה בשם kali linux.
היא לינוקס שהתקינו עליה כלי תקיפה.
← רושמים kalilinux ל downloads ← בתוך הטבלה להוריד kali for vmware.
64bit ← workstation player ← vmware
עושים unzip לתיקייה, ב vmware פותחים מכונה חדשה ולבחור את הקובץ
של kali
username: root
password: toor

maltego -
כלי שמותקן ב kali ועולה המון כסף. צריך להירשם באתר.

individual OSINT
איסוף מידע על אדם אחד. עושים social engineering.
אנשים שבתוך רשת חברתית אפשר להגיע במקסימום 6 צעדים לכל אדם

Exit স্টরমহিলু

আজকে পরিবারের সাথে ঈদ উপলক্ষে বাইরে বেড়াতে যাবার কথা ছিল। কিন্তু আমার মেজাজ খারাপ হয়ে গেল যখন দেখলাম আমার ছোট ভাই ঈদের নতুন জামা পরে বাইরে যাচ্ছে আর আমাকে বলা হচ্ছে বাসায় থাকতে কারণ আমার নাকি পড়া আছে।

বাসায় থাকতে হবে বলে আমি রাগ করে আমার রুমে চলে গেলাম। আমার ছোট ভাই খুবই দুষ্টু। সে সবসময় আমার জিনিস নষ্ট করে। আমি তখন তার খেলনা গাড়ি লুকিয়ে ফেলি। কিছুক্ষণ পর সে কাঁদতে কাঁদতে আমার কাছে আসে। আমি তখন তাকে বলি যে সে যদি আমার কথা শোনে তাহলে আমি তার খেলনা গাড়ি ফিরিয়ে দিব। তখন সে রাজি হয়ে যায়। আমি তখন তার খেলনা গাড়ি ফিরিয়ে দিলাম। তখন সে খুব খুশি হয়ে বাইরে চলে গেল। আমি তখন বুঝতে পারলাম যে ছোট ভাই বোনদের সাথে ভালো ব্যবহার করা উচিত।

পরিশেষে বলা যায় যে, ছোট ভাই বোনদের সাথে ভালো ব্যবহার করা উচিত। তাহলে তারাও আমাদের সাথে ভালো ব্যবহার করবে।

ধন্যবাদ

Nusrat

প্রিয় মাসিমা,

আমি আশা করি আপনি ভালো আছেন। আমার ছোট ভাই খুবই দুষ্টু। সে সবসময় আমার জিনিস নষ্ট করে। আমি তখন তার খেলনা গাড়ি লুকিয়ে ফেলি। কিছুক্ষণ পর সে কাঁদতে কাঁদতে আমার কাছে আসে। আমি তখন তাকে বলি যে সে যদি আমার কথা শোনে তাহলে আমি তার খেলনা গাড়ি ফিরিয়ে দিব। তখন সে রাজি হয়ে যায়। আমি তখন তার খেলনা গাড়ি ফিরিয়ে দিলাম। তখন সে খুব খুশি হয়ে বাইরে চলে গেল। আমি তখন বুঝতে পারলাম যে ছোট ভাই বোনদের সাথে ভালো ব্যবহার করা উচিত।

পরিশেষে বলা যায় যে, ছোট ভাই বোনদের সাথে ভালো ব্যবহার করা উচিত। তাহলে তারাও আমাদের সাথে ভালো ব্যবহার করবে।

ধন্যবাদ

Nusrat

x. কখন মাঝি হয়?

গিয়ে একটি ব্যবসা শুরু করেছে তাহলে সে হয় 'Entrepreneur' এবং তখনই সে উদ্যোক্তা হিসেবে।

* ব্যবসা কি? ব্যবসা হলো একটি প্রক্রিয়া যেটি একজন ব্যক্তি বা প্রতিষ্ঠান লাভের জন্য ক্রয় বিক্রয় করে থাকে।

Ex: Strategy

যেকোনো উদ্যোক্তা যখন কোনো পণ্য বা সেবা তৈরি করে তখন সেটা বাজারে বিক্রি করে। এই বিক্রির জন্য তাকে কিছু কৌশল অবলম্বন করতে হয়। এটাকে বলা হয় Strategy।

8.3.18

Security in the network layer

סופרים את השכבות מלמטה למעלה.
איחדו את שכבות 5-7.
hub - יושב של שכבה 1. מקבל בפורט אחד ויוצא בכל השאר.
switch - יושב של שכבה 2. מעביר. לא מנתב וממתג. מקבל חבילה שצריך להעביר ומעביר ישירות למי שצריך. יש לו טבלה של mac מול פורט. שומר את הכל כדי לבנות את הטבלה. בפעם הראשונה מקבל ושולח לכולם ואז שומר. כל מי שהחבילה לא מיועדת אליו זורק אותה. כל פעם שאין את היעד בטבלה הוא שולח לכולם ומוסיף לטבלה את היעד כשהוא יודע מה הפרטים של היעד כל פעם מציפים את הרשת.
sniffer - יושבים על hub ושומעים מה 2 הצדדים מדברים
אם יושבים על switch זה לא יעבוד ב sniffer ולא נראה מה מדברים.
sniffer - מכשיר או תוכנה ואפשר לשמוע מה כולם אומרים, עובדת על שכבה 2.
צריך להכניס לקונפיגורציה של הכרטיס רשת. יש מצב של הכרטיס רשת פרומיסקוס זה מעלה הכל ואפשר לראות את כל השיחות
אם אינו במוד כזה לא נשאר, צריך מצב פרומיסקוס גם אם היעד הוא ספציפי.
ברשת wifi שומעים הכל, הוא כמו hub.
פרומיסקואוס - מצב שבו שמים את הכרטיס רשת וזה מעלה את כל החבילות גם אם הן לא מיועדות אלי. ע"מ לראות במצב הזה עם סניפר.
אפשר להריץ סניפר ב wireshark.

~~[illegible]~~
evaesdropper
אם אני על hub שומעים מה אומרים, גם אני במצב פרומסקוס שומעים.

adversary
sniffing - התקפה פאסיבית, רק מקשיבים, אי אפשר לגלות אותי.
אין צורך ב hardware מיוחד אבל צריך קונפיגורציה מסויימת.

Poisoning

আমাদের মধ্যে প্রায়শই বিষক্রিয়ার ঝুঁকি থাকে। ১৯৮৪ সালে ভারতের ভোপালে ইউনিয়ন কার্বাইড কারখানা থেকে মিথাইল আইসোসায়ানেট গ্যাস লিক হয়ে হাজার হাজার মানুষ মারা যায়। এটি ছিল শিল্প দুর্ঘটনার ইতিহাসের অন্যতম ভয়াবহ ঘটনা। এই গ্যাস শ্বাসনালী, ফুসফুস এবং চোখের ক্ষতি করে। অনেকে অন্ধ হয়ে যায় বা দীর্ঘমেয়াদি অসুস্থতায় ভোগে। এই ঘটনা আমাদের শিখিয়েছে যে রাসায়নিক পদার্থ সঠিক নিয়ন্ত্রণ ছাড়া কতটা বিপজ্জনক হতে পারে।

বিষক্রিয়া হলো এমন অবস্থা যেখানে কোনো বিষাক্ত পদার্থ শরীরে প্রবেশ করে কোষ ও টিস্যুর ক্ষতি করে। বিষ শরীরে খাদ্য, শ্বাস-প্রশ্বাস বা ত্বকের মাধ্যমে প্রবেশ করতে পারে। বিষের মাত্রা, সংস্পর্শের সময় এবং ব্যক্তির বয়স ও স্বাস্থ্য ফলাফল নির্ধারণ করে। তীব্র বিষক্রিয়া দ্রুত ঘটে এবং জরুরি চিকিৎসা প্রয়োজন হয়। দীর্ঘমেয়াদি বিষক্রিয়া ধীরে ধীরে ক্ষতি করে এবং দীর্ঘদিন অসুস্থতার কারণ হতে পারে।

বিষ বিভিন্ন উপায়ে শরীরে প্রবেশ করতে পারে। খাদ্য ও পানীয় মাধ্যমে বিষ পরিপাকতন্ত্রে প্রবেশ করে। শ্বাস-প্রশ্বাসের মাধ্যমে গ্যাস ও ধোঁয়া ফুসফুসে ঢুকে পড়ে। ত্বক বিষ শোষণ করতে পারে, বিশেষ করে তরল বা গ্যাস আকারে। কীটনাশক ও শিল্প রাসায়নিক ত্বকের মাধ্যমে শোষিত হয়ে ক্ষতি করতে পারে। প্রতিটি প্রবেশ পথের নিজস্ব ঝুঁকি রয়েছে।

বিষ কোষ, টিস্যু ও অঙ্গের ক্ষতি করে। কোষ ঝিল্লি ক্ষতিগ্রস্ত হলে কোষ মারা যায়। এনজাইম নষ্ট হলে বিপাকীয় প্রক্রিয়া ব্যাহত হয়। ডিএনএ ক্ষতিগ্রস্ত হলে ক্যান্সার হতে পারে। লিভার বিষ ফিল্টার করে, তাই লিভার সবচেয়ে ঝুঁকিতে থাকে। কিডনি রক্ত পরিষ্কার করে, ক্ষতিগ্রস্ত হলে কিডনি ব্যর্থতা দেখা দেয়। ফুসফুস শ্বাসনালী ও অ্যালভিওলাস ক্ষতিগ্রস্ত হয়।

বিষক্রিয়ার লক্ষণ বিষের ধরন অনুযায়ী ভিন্ন হয়। বমি, ডায়রিয়া, পেটব্যথা দ্রুত দেখা দেয়। মাথাব্যথা, মাথাঘোরা, অজ্ঞান হওয়া স্নায়ুবিক লক্ষণ। শ্বাসকষ্ট, কাশি, বুকে ব্যথা শ্বাসতন্ত্রের সমস্যা। ত্বকে চুলকানি, ফুসকুড়ি, জ্বালা ত্বকের প্রতিক্রিয়া। হৃৎস্পন্দন দ্রুত বা ধীর হওয়া হৃদযন্ত্রের সমস্যা। জ্বর, ক্লান্তি, ক্ষুধামান্দ্য সাধারণ লক্ষণ।

তীব্র বিষক্রিয়ায় জরুরি চিকিৎসা প্রয়োজন। বিষ খেলে বমি করানো যেতে পারে, তবে ডাক্তারের পরামর্শ নিতে হবে। কয়লা বিষ শোষণ করে। দুধ বা পানি দিয়ে বিষ পাতলা করা যায়। হাসপাতালে গ্যাস্ট্রিক ওয়াশ, ডায়ালিসিস, ওষুধ প্রয়োগ করা হয়। দ্রুত হাসপাতালে নেওয়া জীবন বাঁচায়। প্রাথমিক চিকিৎসা ও হাসপাতাল চিকিৎসা উভয়ই গুরুত্বপূর্ণ।

বিষ এড়ানোর জন্য সতর্কতা জরুরি। রাসায়নিক পদার্থ সঠিক পাত্রে রাখা উচিত। খাদ্য সঠিক তাপমাত্রায় সংরক্ষণ করা প্রয়োজন। কীটনাশক ব্যবহারে সতর্কতা অবলম্বন করা উচিত। শিল্পে নিরাপত্তা নিয়ম মেনে চলা জরুরি। শিশুদের নাগালের বাইরে বিষ রাখা উচিত। সচেতনতা দুর্ঘটনা কমায়।

ফ্লোচার্ট (Flowchart) হচ্ছে কোনো প্রোগ্রাম, জটিল অংশ কিংবা অন্য কোনো প্রসেস-এর বিভিন্ন ধাপ ও তথ্যের প্রবাহ দেখানোর জন্য ব্যবহৃত একটি চিত্রমূলক উপস্থাপনা। এটি প্রোগ্রামার বা বিশ্লেষকদের জটিল লজিক বুঝতে, ডিবাগ করতে এবং অন্যদের কাছে ব্যাখ্যা করতে সাহায্য করে।

ফ্লোচার্টের মূল উপাদান:

- টার্মিনাল: প্রোগ্রামের শুরু বা শেষ বোঝায়।
- প্রসেস: কোনো গণনা বা কাজ বোঝায়।
- ডিসিশন: শর্ত যাচাই করে (হ্যাঁ/না)।
- ইনপুট/আউটপুট: ডেটা গ্রহণ বা প্রদর্শন।
- সংযোগ: তীর দিয়ে ধাপগুলোর মধ্যে সংযোগ।

ফ্লোচার্ট কেন ব্যবহার করা হয়?

- প্রোগ্রাম ডিবাগিং: কোথায় ভুল হচ্ছে তা দ্রুত বোঝা যায়।
- ডকুমেন্টেশন: ভবিষ্যতে কোড বুঝতে সাহায্য করে।
- টিম কমিউনিকেশন: প্রোগ্রামার ও নন-প্রোগ্রামারদের মধ্যে ধারণা শেয়ার করা।
- অ্যালগরিদম ডিজাইন: কোড লেখার আগে পরিকল্পনা করা।

উদাহরণ: একটি সংখ্যা জোড় নাকি বিজোড় তা যাচাই করার ফ্লোচার্ট

ধাপ ১: শুরু (টার্মিনাল)

ধাপ ২: ইনপুট নেওয়া (ইনপুট/আউটপুট)

ধাপ ৩: সংখ্যাটি ২ দিয়ে ভাগ করা হচ্ছে কি না? (ডিসিশন)

ধাপ ৪: হ্যাঁ হলে "জোড়", না হলে "বিজোড়" প্রিন্ট করা (প্রসেস)

ধাপ ৫: শেষ (টার্মিনাল)

ফ্লোচার্ট তৈরির টুল: Draw.io, Lucidchart, Microsoft Visio, Figma, Excel/Word-এর Shapes ফিচার।

ফ্লোচার্ট বনাম ফ্লো ডায়াগ্রাম: ফ্লো ডায়াগ্রাম সাধারণ প্রক্রিয়া দেখায়, কিন্তু ফ্লোচার্ট বিশেষভাবে প্রোগ্রামিং লজিক দেখায়।

ফ্লোচার্ট (Flowchart) হচ্ছে একটি প্রোগ্রাম, জটিল অংশ কিংবা অন্য কোনো প্রসেস-এর বিভিন্ন ধাপ ও তথ্যের প্রবাহ দৃশ্যমান করার জন্য ব্যবহৃত একটি চিত্রমূলক উপস্থাপনা। এটি প্রোগ্রামার বা বিশ্লেষকদের জটিল লজিক বুঝতে, ডিবাগ করতে এবং অন্যদের কাছে ব্যাখ্যা করতে সাহায্য করে।

ফ্লোচার্টের মূল উপাদান:

- টার্মিনাল: প্রোগ্রামের শুরু বা শেষ বোঝায়।
- প্রসেস: কোনো গণনা বা কাজ বোঝায়।
- ডিসিশন: শর্ত যাচাই করে (হ্যাঁ/না)।
- ইনপুট/আউটপুট: ডেটা গ্রহণ বা প্রদর্শন।
- সংযোগ: তীর দিয়ে ধাপগুলোর মধ্যে সংযোগ।

ফ্লোচার্ট কেন ব্যবহার করা হয়?

- প্রোগ্রাম ডিবাগিং: কোথায় ভুল হচ্ছে তা দ্রুত বোঝা যায়।
- ডকুমেন্টেশন: ভবিষ্যতে কোড বুঝতে সাহায্য করে।
- টিম কমিউনিকেশন: প্রোগ্রামার ও নন-প্রোগ্রামারদের মধ্যে ধারণা শেয়ার করা।
- অ্যালগরিদম ডিজাইন: কোড লেখার আগে পরিকল্পনা করা।

উদাহরণ: একটি সংখ্যা জোড় নাকি বিজোড় তা যাচাই করার ফ্লোচার্ট

ধাপ ১: শুরু (টার্মিনাল)

ধাপ ২: ইনপুট নেওয়া (ইনপুট/আউটপুট)

ধাপ ৩: সংখ্যাটি ২ দিয়ে ভাগ করা হচ্ছে কি না? (ডিসিশন)

ধাপ ৪: হ্যাঁ হলে "জোড়", না হলে "বিজোড়" প্রিন্ট করা (প্রসেস)

ধাপ ৫: শেষ (টার্মিনাল)

ফ্লোচার্ট তৈরির টুল: Draw.io, Lucidchart, Microsoft Visio, Figma, Excel/Word-এর Shapes ফিচার।

ফ্লোচার্ট বনাম ফ্লো ডায়াগ্রাম: ফ্লো ডায়াগ্রাম সাধারণ প্রক্রিয়া দেখায়, কিন্তু ফ্লোচার্ট বিশেষভাবে প্রোগ্রামিং লজিক দেখায়।

ইনপুট সিগন্যাল কিছু ফ্রিকোয়েন্সি।

2kHz এবং এটাকে আরেকটা প্রবাহ করতে ফ্রিকোয়েন্সি এবং 4kHz

H কে এটা টেবিলে কখন কখন যাবে দেখি

2kHz এ θ এর মধ্যে সাব্ধারে আরেকটা প্রবাহ করতে ফ্রিকোয়েন্সি

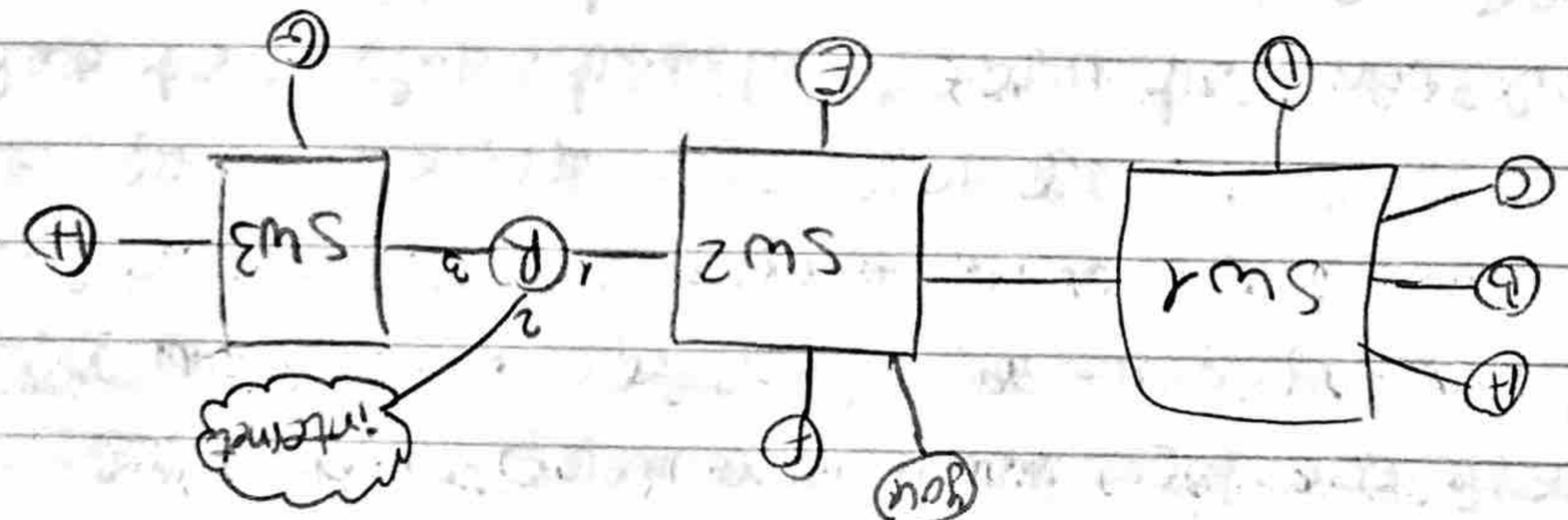
* এই সে প্রবাহের সাথে করতে হলে সেন্স, মধ্যে ৩ এর সেন্সর এটা কি করে যে

করতে হবে।

এখন 0.8 θ এর মধ্যে আরেকটা প্রবাহ করতে ফ্রিকোয়েন্সি H-এ θ এর মধ্যে করতে হবে। এবং এটা করতে হলে H-এ θ এর মধ্যে করতে হবে।

এটা করতে হলে H এর মধ্যে করতে হবে। এবং এটা করতে হলে H এর মধ্যে করতে হবে।

এখন আমরা দেখব কিভাবে এই সার্কিটটা কাজ করে। ধরা যাক ইনপুট সিগন্যাল এসেছে 2kHz এবং 4kHz ফ্রিকোয়েন্সি। এখন এটা যদি সুইচ S1 এর মধ্যে আসে তাহলে এটা সুইচ S2 এর মধ্যে আসবে। এখন এটা যদি সুইচ S2 এর মধ্যে আসে তাহলে এটা সুইচ S3 এর মধ্যে আসবে। এখন এটা যদি সুইচ S3 এর মধ্যে আসে তাহলে এটা সুইচ S4 এর মধ্যে আসবে।



ফিল্টার-B, সুইচ-S3

2f-4f এর মধ্যে ফ্রিকোয়েন্সি থাকলে এটা ফিল্টার করে দেবে। এখন আমরা দেখব কিভাবে এই সার্কিট কাজ করে। ধরা যাক ইনপুট সিগন্যাল এসেছে 2kHz এবং 4kHz ফ্রিকোয়েন্সি। এখন এটা যদি সুইচ S1 এর মধ্যে আসে তাহলে এটা সুইচ S2 এর মধ্যে আসবে। এখন এটা যদি সুইচ S2 এর মধ্যে আসে তাহলে এটা সুইচ S3 এর মধ্যে আসবে। এখন এটা যদি সুইচ S3 এর মধ্যে আসে তাহলে এটা সুইচ S4 এর মধ্যে আসবে।

প্রতিটি ত্রিভুজের ক্ষেত্রে H এর মান বসিয়ে ক্ষেত্রফল নির্ণয় করা।

DHCP

IP সাবনেট মাস্ক এবং DHCP এর মধ্যে পার্থক্য:

DNS, Subnet mask, IP: ডোমেইন নেমের IP অ্যাড্রেস বের করতে DNS ব্যবহার করা হয়। IP অ্যাড্রেসকে সাবনেট মাস্ক দিয়ে ভাগ করে নেটওয়ার্ক এবং হোস্ট আলাদা করা হয়। DHCP দিয়ে স্বয়ংক্রিয়ভাবে IP অ্যাড্রেস, সাবনেট মাস্ক, ডিফল্ট গেটওয়ে এবং DNS সার্ভার নির্ধারণ করা হয়।

Subnet mask: IP অ্যাড্রেসকে নেটওয়ার্ক এবং হোস্ট অংশে ভাগ করে।

DNS

ডোমেইন নেমকে IP অ্যাড্রেসে রূপান্তর করে। DNS সার্ভার ডোমেইন নেম রিকোয়েস্ট গ্রহণ করে এবং সংশ্লিষ্ট IP অ্যাড্রেস ফিরিয়ে দেয়। ডোমেইন নেম হিরার্কি অনুসরণ করে কাজ করে। DNS সার্ভার ক্যাশিং ব্যবহার করে দ্রুত রেজাল্ট দেয়।

DHCP

DHCP হোস্টকে স্বয়ংক্রিয়ভাবে IP অ্যাড্রেস, সাবনেট মাস্ক, ডিফল্ট গেটওয়ে এবং DNS সার্ভার নির্ধারণ করে। DHCP ছাড়া ম্যানুয়ালি IP অ্যাড্রেস কনফিগার করা হয়। DHCP ডিভাইস ব্যবস্থাপনা সহজ করে।

ব্যবহার: DHCP ব্যবহার করে IP অ্যাড্রেস বণ্টন করা হয়। Broadcast: নেটওয়ার্কের সবার কাছে মেসেজ পাঠানো হয়। Unicast: নির্দিষ্ট একটি ডিভাইসকে মেসেজ পাঠানো হয়। Broadcast নেটওয়ার্ক ট্রাফিক বাড়ায়। Unicast দক্ষ এবং নিরাপদ।

Broadcast: নেটওয়ার্কের সবার কাছে মেসেজ পাঠানো হয়। Unicast: নির্দিষ্ট একটি ডিভাইসকে মেসেজ পাঠানো হয়। Broadcast নেটওয়ার্ক ট্রাফিক বাড়ায়। Unicast দক্ষ এবং নিরাপদ।

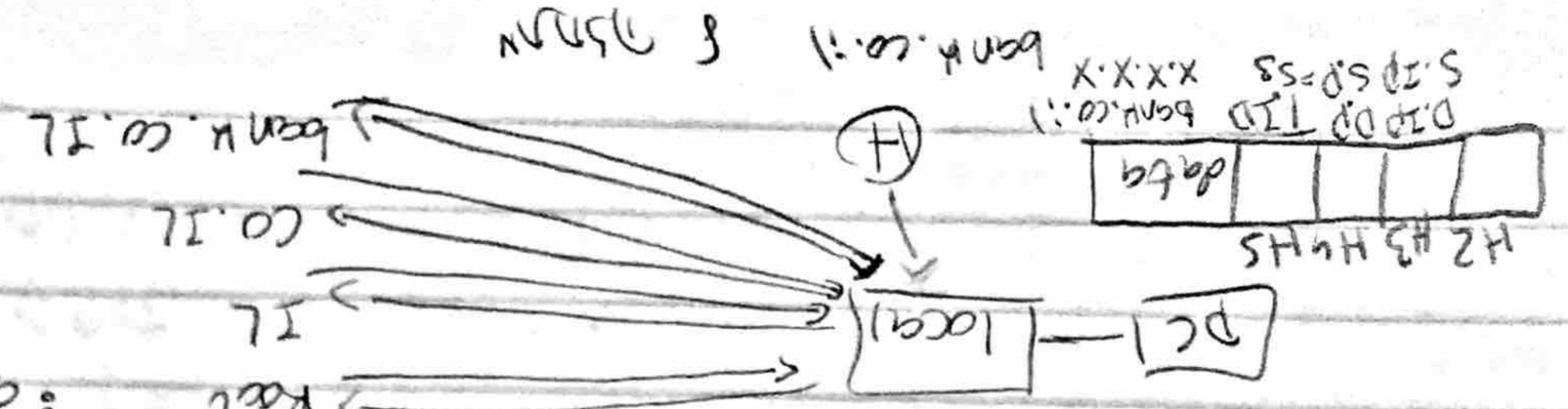
ব্যবহার: DHCP ব্যবহার করে IP অ্যাড্রেস বণ্টন করা হয়। Broadcast: নেটওয়ার্কের সবার কাছে মেসেজ পাঠানো হয়। Unicast: নির্দিষ্ট একটি ডিভাইসকে মেসেজ পাঠানো হয়। Broadcast নেটওয়ার্ক ট্রাফিক বাড়ায়। Unicast দক্ষ এবং নিরাপদ।

ব্যবহার: DNS ব্যবহার করে ডোমেইন নেম রিজলভ করা হয়। Recursive: DNS সার্ভার নিজে সবার হয়ে কাজ করে। Iterative: ক্লায়েন্ট নিজে একটি একটি সার্ভার থেকে তথ্য সংগ্রহ করে। Recursive দ্রুত এবং সহজ। Iterative নিয়ন্ত্রণ বেশি।

Recursive: DNS সার্ভার নিজে সবার হয়ে কাজ করে। Iterative: ক্লায়েন্ট নিজে একটি একটি সার্ভার থেকে তথ্য সংগ্রহ করে। Recursive দ্রুত এবং সহজ। Iterative নিয়ন্ত্রণ বেশি।

ব্যবহার: DNS ব্যবহার করে ডোমেইন নেম রিজলভ করা হয়। Recursive: DNS সার্ভার নিজে সবার হয়ে কাজ করে। Iterative: ক্লায়েন্ট নিজে একটি একটি সার্ভার থেকে তথ্য সংগ্রহ করে। Recursive দ্রুত এবং সহজ। Iterative নিয়ন্ত্রণ বেশি।

ואם ה TLD לא ידוע הוא משנה רמה מתחת לו' שמגיים.
כשיש תשובה - הלקוח רושם אצלו ומעביר את התשובה הלאה.
authoritive ההשרת והתשובה נקראים resource record, הוא יכול
להכיל record אחד שיש שאני שלך ותשאל אותי, כמה הבאה לא פונה
לחני אבנה.
2. רקורסיבי - פחות משתמשים בה. אני שואל את ה root ומאקום לקבל
הוא שואל וכה ינחה בהיררכיה ובסוף ה root מחזיר לי תשובה.
אם נצליח לגרום לקוחאל במקום אחרי השרתים כה אומר שכל הלקוחות
שמשתמשים בלקוחאל הזה יפנו אלי. נצליח לפשות הרעלה לכמה לקוחאל.
צריך להכיר את ה header של DNS.
DNS הוא בשכבה 5. כל מעל TCP בפורט 53. UDP הוא פשוט
כמה ו שלות ב header כמה סוגים של קוד ושאלה.
בשכבה 5 לא בהשרת DNS יש transaction ID בכל בקשה מוספת
וכל תשובה צריכה להתאים למספר.
פורט ברירת- 53, פורט שלח- רנדומלי.



এনক্রিপশন সমস্যাটি: প্রথমেই আমরা ধরে নিচ্ছি যে ব্লকের ডাটা এনক্রিপ্টেড। এখন প্রশ্ন হলো এই ডাটা কি সত্যিই ব্লকের মালিকের কাছ থেকে এসেছে? নাকি কোনো হ্যাকার এসে ডাটা বদলে দিয়েছে? এটাকে বলা হয় ইন্টিগ্রিটি (Integrity) সমস্যা।

সমাধান হিসেবে আমরা একটি ডিজিটাল সিগনেচার ব্যবহার করতে পারি। ব্লকচেইনে প্রতিটি ব্লক তৈরি করার সময় মাইনার বা ভ্যালিডেটর একটি ডিজিটাল সিগনেচার যুক্ত করে। এটি একটি গাণিতিক ফর্মুলা যা ব্লকের ডাটা এবং মাইনারের প্রাইভেট কি ব্যবহার করে তৈরি হয়। এর সুবিধা হলো, কেউ যদি ডাটা পরিবর্তন করে, তাহলে সিগনেচার আর মিলবে না, আর সবাই বুঝে যাবে যে ডাটা নষ্ট হয়েছে।

তবে এখানে একটি সমস্যা আছে। ধরুন কোনো হ্যাকার পুরো ব্লকটিই মুছে ফেলে এবং নতুন করে একটি ব্লক তৈরি করে, যেটার সিগনেচার ঠিক আছে, তাহলে কি সমস্যা হবে? এটাকে বলা হয় রিপ্লে অ্যাটাক (Replay Attack)। এর সমাধান হিসেবে আমরা টাইমস্ট্যাম্প ব্যবহার করতে পারি। প্রতিটি ব্লকে একটি টাইমস্ট্যাম্প থাকে, এবং নেটওয়ার্ক সময়ের একটি সীমা নির্ধারণ করে দেয়, যেমন ১০ মিনিট। এর বাইরে কোনো ব্লক গ্রহণযোগ্য নয়।

ডাবল স্পেন্ডিং (Double Spending) সমস্যা: ধরুন আমার কাছে ১০ টাকা আছে। আমি একজনকে ১০ টাকা দিলাম এবং একই টাকা আরেকজনকে দিলাম। এটা ব্লকচেইনে কীভাবে সম্ভব? এর সমাধান হিসেবে প্রুফ অফ ওয়ার্ক (Proof of Work) বা প্রুফ অফ স্টেক (Proof of Stake) ব্যবহার করা হয়। এর মাধ্যমে নেটওয়ার্কের অধিকাংশ নোড একমত হয় যে কোন লেনদেনটি সঠিক।

৫১% অ্যাটাক: যদি কোনো ব্যক্তি বা গোষ্ঠী নেটওয়ার্কের ৫১% ক্ষমতা নিয়ন্ত্রণ করে, তাহলে সে ইচ্ছামতো ব্লকচেইন পরিবর্তন করতে পারে। এটি ব্লকচেইনের একটি বড় দুর্বলতা। এর সমাধান হিসেবে বিকেন্দ্রীকরণ (Decentralization) গুরুত্বপূর্ণ।

Q.2 Hash function: হ্যাশ ফাংশন হলো একটি গাণিতিক ফাংশন যা যেকোনো ইনপুট ডাটাকে একটি ফিক্সড সাইজের আউটপুটে রূপান্তরিত করে। এটি একমুখী (One-way) ফাংশন, অর্থাৎ আউটপুট থেকে ইনপুট বের করা প্রায় অসম্ভব।

ব্লকচেইনে ব্যবহৃত হ্যাশ ফাংশনের বৈশিষ্ট্য:

১. নির্ধারণযোগ্য (Deterministic): একই ইনপুট দিলে সবসময় একই আউটপুট পাওয়া যাবে।

২. দ্রুত গণনা (Fast Computation): যেকোনো ইনপুট দ্রুত হ্যাশ করা যায়।

৩. ইনপুট পরিবর্তনে বড় পরিবর্তন (Avalanche Effect): ইনপুটের সামান্য পরিবর্তনে আউটপুট সম্পূর্ণ আলাদা হয়ে যায়।

৪. কলিশন রেজিস্ট্যান্স (Collision Resistance): দুটি আলাদা ইনপুটের একই হ্যাশ আউটপুট হওয়া খুবই কঠিন।

৫. প্রিইমেজ রেজিস্ট্যান্স (Preimage Resistance): হ্যাশ থেকে মূল ডাটা বের করা অসম্ভব।

৬. ফিক্সড সাইজ আউটপুট: যেমন SHA-256 সবসময় ২৫৬ বিট আউটপুট দেয়।

ব্লকচেইনে ব্যবহৃত হ্যাশ ফাংশন: SHA-256, Keccak-256 (Ethereum), Scrypt (Litecoin)।

Q.3 Proof-Security in the Bitcoin Ledger

Mining is SHA-256.

בתמונה שהראתה שלח יש data ויש header של שכבה 2,3,4,5.
ב data יש תשובת DNS, bank.co.il יושבת בכתובת X.X.X.X
בתוך ה data יש TID שבתוך headers, ב-4 יש פורטים
destination ו source, ב-3 יש IP יעד ומקור.
הבעיה בפורטים וב TID שצריך לנחש אותם. TID-transaction ID
פורט עד כמה משהו קבוע ואפשר לנחש אותו.
TID זה מספר רץ, מריצים ומנסים לנחש את המספר, מריצים את כל המספרים.
פורט מקור תמיד יהיה 53.
נקרא קמינסקי attack.
מה עושים? נשים פורט רנדומלי במקום קבוע במקום ש TID יהיה מספר רץ הוא יהיה רנדומלי.
אחרי שהרעלנו אתר, אפשר להגיע לשרת התילים שלו. ברגע שנעביר לאתר שלי ולא לאתר של הבנק, אין את המנעול הירוק בכתובת של האתר (SSL).
מישהו נותן certificate שיש את המנעול הירוק וצריך לגשת אליו, מי שנותן את האישור מחפש whois ושולח את האישור למייל שלהם, אם הצלחתי להגיע לשרת של המאשר אוכל לקבל את המנעול הירוק.
ככל שיש יותר ביטים ב TID יותר קשה לתקוף.

Network attack

DoS

מונע משרת מתת שירות

1. פיזי - פגיעה פיזית ברשת. ניתוק ראוטר. פגיעה ברשת או במשהו שבלעדיו הוא לא יכול לתפקד.

2. קונפיגורציה - להיכנס לרגיסטרי ולהפוך משהו, שיעבירו למקום אחר במקום ל default gateway קונפיגורציה.

3. ניסיון לנצל משאבים יקרים כמו כוח עיבוד, זיכרון, כמה חבילות יכול להוציא.

כשממלאים את ה bandwith אפשר לשלוח מלא חבילות של UDP, נקרא UDP packet storm.

ping flood - שליחת הרבה pings מלא מחשבים.

שאלה: מה קורה אם שולחים חבילה לפורט שמאזינים בו? המקבל עונה.

מה קורה אם שולחים חבילה לפורט שלא מאזין בו?
יענה בחבילה ICMP שהפורט לא ניתן להשגה.

רוצים להפסיק את השרת.

אם רוצים לשנות משהו משנים קונפיגורציות (רגיסטרי, טבלאות ניתוב).

אפשר להתגונן לבנאי ולפגיעה בכמה הפיזית, לא בהרבה בשרתים; אולי בענקים כי קשה להגיע לשרתים.

DoS עושים לבד?
שמים חומת אש אם מישהו שולח הרבה חבילות וחוסמים אותו.

DDoS

שולחים מהרבה כתובות IP, אפשר עם בוטים, זומבים, חברים, נכנסים ל host ומקבלים מחשבים עם IP שונים ושולחים חבילות לשרת. botnet (רשת רובוטים). ולוקחים מחשב C&C ששולט על הרובוטים.
הרעיון: מתכננים התקפה ומפעילים אותה, כולם זורקים בו זמנית ועושים את ההתקפה. C&C עצמו צריך להגיע לכולם במכה שיתחילו לפעול.
התקשורת בין התוקף ל controller היא ב TCP, בין ה controller לבוטים גם היא ב TCP. התוקף מעביר את הפקודה ב TCP וה controller מעביר לבוטים ב UDP. אין צורך ל TCP מול כל אחד.

ICMP

פרוטוקול שנשלח כ ICMP.

מדד בקרת תקשורת שגיאות.

internet control message protocol

IP and TCP

בנוסף יש UDP.

שקף 26.

מעל משרד handshake 3 way שמבצע בתחילת תקשורת.

תקשורת שולחת לשרת.

בשרת יש מטמון מלבנים.

אחרי שקיבלתי את ה מעל אני במלבן wait

שאלה: איך מבצע handshake way 3 מתי הבאתם?

ping-of death - פינג מה תכולת echo request. לקחו תכולה

גדולה נשלחו פינג אחד

כל מערכת הפעלה יודעת לשלוח פינג בגדלים.

teardrop - תכולה שמשרתים את תכולה הקודמת. לבתוך ב offset

מה הבן יתר שלרוס את התכולה הקודמת ואל הרשת נתקלת

בעומס רשתות משהו שאי.

bonk - אותו כמו אבל נרשם מס' שלם

פתרון: implementation TCP תקין יתר, שמרים מערכת הפעלה

land - שולח מעל שרתים ומקבל אותו לבד (של התקבל) והוא

יכנס ל loop.

smurf - שולחים פינג עם יעד broadcast ומאקדר שמים את התקבל

נגד מעמים אלו.

מתבטאים חוסר סטרוגות בתקשורת.

פתרון: לא מאפשר לרכנים תכולות ב broadcast.

Exam Roll Finding Dot

প্রোগ্রাম 2 ঘ. ডাটা ও
প্রসেসিং ফাইল পাঠকের নাম ও রোল নম্বর -২ প্রোগ্রাম
যে ফাইল পাঠকের নাম ও রোল নম্বর -২ প্রোগ্রাম
এর আউটপুট হিসেবে তৈরি করেছে সেই ফাইলটি
পড়ে তার ডাটা স্ক্রিনে প্রিন্ট করবে।

```
FILE *fp;
int roll, n;
char name[20];
fp = fopen("roll.dat", "r");
while (fscanf(fp, "%d %s", roll, name) != EOF)
    printf("%d %s\n", roll, name);
fclose(fp);
```

ফাইল থেকে ডাটা পড়ার জন্য fscanf() ব্যবহার করা হয়েছে।
fscanf() ফাংশনটি scanf() এর মতো কাজ করে কিন্তু
কীবোর্ড থেকে ইনপুট না নিয়ে ফাইল থেকে ইনপুট নেয়।
fscanf() ফাংশনের প্রথম আর্গুমেন্ট হিসেবে ফাইল
পয়েন্টার দেওয়া হয়েছে।
fscanf() ফাংশনটি ফাইলের শেষে পৌঁছালে EOF
রিটার্ন করে। EOF হলো একটি স্পেশাল কনস্টান্ট
যা ফাইলের শেষ নির্দেশ করে।
fscanf() ফাংশনের রিটার্ন ভ্যালু চেক করে
লুপ চালানো হয়েছে।
ফাইল ক্লোজ করা হয়েছে fclose() ফাংশন দিয়ে।
ফাইল ক্লোজ না করলে মেমরি লিক হতে পারে
এবং ডাটা সঠিকভাবে সেভ নাও হতে পারে।
ফাইল ওপেন করা হয়েছে "r" মোডে (read mode)
যাতে শুধু পড়া যায়, লেখা যায় না।
ফাইল পয়েন্টার fp ব্যবহার করে ফাইল থেকে
ডাটা পড়া হয়েছে।
ফাইল থেকে ডাটা পড়ার সময় ফরম্যাট স্পেসিফায়ার
ব্যবহার করা হয়েছে যেমন %d, %s ইত্যাদি।
ফাইল থেকে ডাটা পড়ার সময় ফরম্যাট স্পেসিফায়ার
ব্যবহার করা হয়েছে যেমন %d, %s ইত্যাদি।
ফাইল থেকে ডাটা পড়ার সময় ফরম্যাট স্পেসিফায়ার
ব্যবহার করা হয়েছে যেমন %d, %s ইত্যাদি।

נשים proxy על מי שרוצים להגן עליו.

דוגמאות.

1. Coock אוגוסט 2003. MS blast. אף אחד לא יכל לעצור את הוירוס.

estonia attack

נעשה ב 2007.

בוטנטים

מ syn TCP ר

12 מתקפות של 70-95 Mbps במשך 10 שבועות.

הם התנתקו מהרשת.

betcris - אתר הימורים שהותקף ע"י 50 אלף מחשבים, ושלחו 2GB פר שנייה של syn (בשנת 2003)

detection methods

syn-fin behavior - מס' ה syn צריך להיות בערך כמו מס' ה fin. כמות הפתיחות וכמות הסגירות צריכה להיות בערך זהה.

preventing DoS

סינקוקי-cookie - שומר מידע, נמצא בפרוטוקול עצמו.

רוצים להוציא סינקוקי cookie ב TCP.

השרת צריך להיות לא מודע ולא יודע ומתאים למה שכולם מדברים איתו.

ברגיל נשלח syn ונפתח מכונת מצבים.

ב sequence לא נשים סתם איזה מס' מתחילים אלא את המס' הבא.

כל מערכת הפעלה מתחילה מאיזה מס' שבא לה

נחפש את המס' הזה כדי לדעת איזה מערכת מולי.

אם אין את המס' נשים איזה מס' שרוצים ונשים שם סינקוקי.

יחזור אלי ACK עם המס' ששלחתי +1, אם נוריד 1, נדע את כל הפרטים.

הבעיה הייתה מכונת מצבים, ונבטל אותה אם נשמור נתונים ונחזיר אותם ללקוח, המס' שב sequence לא מוגבר, נחשב את ה סינקוקי מחדש ונשלח, כך אם רגיל יחזיר סינקוקי cookie +1, מורידים 1 מהמס' ומקבל את המידע.

HTTP Header

এইচটিটিপি হেডার ক্লায়েন্ট এবং সার্ভার যখন রেসপন্স বা রিকোয়েস্ট পাঠায় তখন এর সাথে কিছু অতিরিক্ত তথ্য পাঠায় যা রিকোয়েস্ট বা রেসপন্সের বিষয়বস্তু সম্পর্কে বলে। এগুলোকে বলা হয় HTTP হেডার।

Empty connection Error

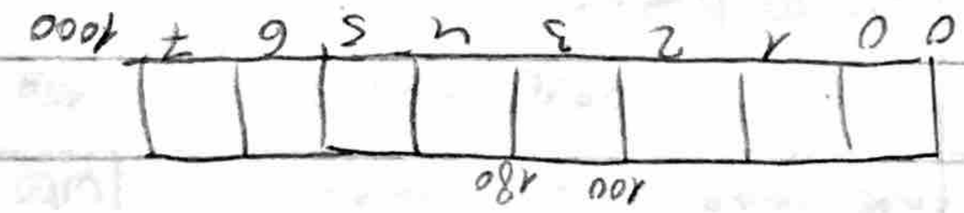
কোনো কানেকশন এরর দেখা দিলে ব্রাউজার সার্ভারকে জানায় যে কানেকশন বন্ধ হয়ে গেছে বা তৈরি করা যায়নি। এটি সাধারণত নেটওয়ার্ক সমস্যা, সার্ভার ক্রাশ বা ফায়ারওয়াল ব্লক করলে হতে পারে।

TCP connection Error

TCP কানেকশন এরর তখন ঘটে যখন ক্লায়েন্ট সার্ভারের সাথে সংযোগ স্থাপন করতে ব্যর্থ হয়। এটি সাধারণত নেটওয়ার্ক সমস্যা, ফায়ারওয়াল ব্লক বা সার্ভার ডাউন থাকলে হতে পারে।

ভেরিয়েবল

ধরি একটি ফাংশন আছে যা ৫-৫ এর মধ্যে একটি র‍্যান্ডম সংখ্যা দেয়, যা ক্রেডিট কার্ডের শেষ ৪ ডিজিট। ৫-২ এর বাইরে যদি কোনো সংখ্যা আসে তখন ফাংশন আবার কল করে যতক্ষণ না ৫-২ এর মধ্যে সংখ্যা আসে। ধরি ফাংশনটি ৫, ৬, ৭, ৮, ৯, ১০ এই সংখ্যাগুলো রিটার্ন করে। এখন আমরা ৪-ডিজিট পাসওয়ার্ড তৈরি করতে চাই।



১০০০ থেকে ১৮০ বাদ দিলে, ৪ অঙ্কের সব সংখ্যা পাওয়া যাবে। এখন ৪ অঙ্কের সংখ্যা থেকে ৩ অঙ্কের সংখ্যা বাদ দিলে ৪ অঙ্কের সংখ্যা পাওয়া যাবে।

ডিফল্ট পোর্ট

ডিফল্ট পোর্ট হলো সেই পোর্ট যা কোনো প্রোটোকল ব্যবহার করে। যেমন HTTP ডিফল্ট পোর্ট ৮০, HTTPS ডিফল্ট পোর্ট ৪৪৩, FTP ডিফল্ট পোর্ট ২১, SMTP ডিফল্ট পোর্ট ২৫, DNS ডিফল্ট পোর্ট ৫৩ ইত্যাদি। ডিফল্ট পোর্ট ব্যবহার করলে URL-এ পোর্ট উল্লেখ করতে হয় না।

প্রসিডিউরাল প্রোগ্রামিং

প্রসিডিউরাল প্রোগ্রামিং হলো এমন একটি প্রোগ্রামিং পদ্ধতি যেখানে ডেটা এবং ফাংশন আলাদা থাকে। এটি সাধারণত ছোট প্রজেক্টের জন্য ভালো কাজ করে। বড় প্রজেক্টে ডেটা ও ফাংশন আলাদা থাকায় কোডের রক্ষণাবেক্ষণ কঠিন হয়ে যায়। উদাহরণ হিসেবে C, Pascal, Fortran, COBOL ইত্যাদি ভাষা প্রসিডিউরাল প্রোগ্রামিং ব্যবহার করে।

Dos by connection reset

প্রথমে নেটওয়ার্ক চেক করে নিন। হয়তো কোনোভাবে আমার নেট সমস্যা করছে। এরপর
প্রয়োজন হলে

টেল্ নেট কানেকশন স্লোয়িং

প্রায়ই এটা আসলে এক ধরনের
প্রটেক্টেড মোড হিসেবে কাজ করে (এটি কোনো ফিচার কিংবা ফাইল নয়)

finding	source	reconnaissance	ID
כל הכתובות רשומות תחת AS החיצוניות של החברה	ASN-mxtoolbox		1
domain name: liveperson.com registry domain ID: 2090814_domain_com-VRSN registrar IANA ID: 2	who is		2
net range: 199.187.116.0-199.187.119.255 org ID: livepe address: 475 10th avenue, 5th floor city: new york state prov: NY postal code: 10018	who is 199.187.116.00		3
תישק בקלים מסך			
56(84) bytes of data	ping		4
איזה אימיילים של עובדים קיימים (בלים מסך)	the harvester		5
איזה תפקידים של עובדים ב linkedin (בלים מסך)	the harvester linkedin		6
hostname IPs	the harvester liveperson.net		7

סניפר wireshark מאזין חבילות.
שמים סניפר ב-2 המחשבים ורואים מה עובר ומה לא ומאיזה מחשב.
הבעיה- המחשב המותקף לא מכיר את הכתובת IP של המחשב המתקיף
ופה הבעיה, השנוי ל NAT.

22.6.24 HTTP Flood Case

Post-1 পোস্ট করে PHP ও ডাটা
প্রোসেস করার জন্য ব্যবহার করা হয় কিছু সার্ভার
সার্ভারে সরাসরি পোস্ট রিকোয়েস্ট পাঠানো হচ্ছে

Load Balancing Strategy

ক্লায়েন্ট রিকোয়েস্ট এলে লোড ব্যালান্সার সেই রিকোয়েস্ট
সার্ভারগুলোর মধ্যে সমানভাবে ভাগ করে দেয়।
এতে কোনো একটি সার্ভার অতিরিক্ত চাপ নিতে পারে না।
ফলে সিস্টেম দ্রুত এবং স্থিতিশীল থাকে।

HTTP Complicated attack

সাধারণ HTTP অ্যাটাক সাধারণত একটি URL বারবার রিকোয়েস্ট পাঠায়।
কিন্তু এই অ্যাটাকে রিকোয়েস্টগুলো দেখতে বৈধ মনে হয়।
সার্ভার বুঝতে পারে না এগুলো আক্রমণ।
ফলে সার্ভার ধীরে ধীরে ধীর হয়ে যায় বা ক্রাশ করে।

Tcp connection reset

সাধারণ ট্রাফিকের তুলনায় কোনো একটি IP থেকে
(অথবা কিছু IP থেকে) খুব দ্রুত কানেকশন রিসেট করা হচ্ছে।
এটা সাধারণত DDoS অ্যাটাকের আগে বা পরে হতে পারে।
প্রক্সি সার্ভার, লোড ব্যালান্সার বা ফায়ারওয়াল ব্যবহার করে
এই অ্যাটাক কমানো যায়।

Discussion questions

১ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং এ প্রতিটি রাউটার তার প্রতিবেশী রাউটারগুলোর সাথে তথ্য বিনিময় করে এবং সেই তথ্য ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি বেলমান-ফোর্ড অ্যালগরিদম ব্যবহার করে কাজ করে।

২ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার পুরো নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ বের করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৩ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

৪ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৫ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

Discussion questions

১ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং এ প্রতিটি রাউটার তার প্রতিবেশী রাউটারগুলোর সাথে তথ্য বিনিময় করে এবং সেই তথ্য ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি বেলমান-ফোর্ড অ্যালগরিদম ব্যবহার করে কাজ করে।

২ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার পুরো নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ বের করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৩ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

৪ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৫ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

DNS কমপ্লিকেশন থেকে

ডোমেন নেমিং সিস্টেম হল একটি বিতরণকৃত ডেটাবেস যা ডোমেন নেম এবং IP অ্যাড্রেসের মধ্যে মানচিত্র তৈরি করে। এটি ইন্টারনেটের একটি গুরুত্বপূর্ণ উপাদান। এর মাধ্যমে ব্যবহারকারীরা সংখ্যার পরিবর্তে নাম ব্যবহার করে ওয়েবসাইট অ্যাক্সেস করতে পারে।

DNS কমপ্লিকেশন থেকে

ডোমেন নেমিং সিস্টেম হল একটি বিতরণকৃত ডেটাবেস যা ডোমেন নেম এবং IP অ্যাড্রেসের মধ্যে মানচিত্র তৈরি করে। এটি ইন্টারনেটের একটি গুরুত্বপূর্ণ উপাদান। এর মাধ্যমে ব্যবহারকারীরা সংখ্যার পরিবর্তে নাম ব্যবহার করে ওয়েবসাইট অ্যাক্সেস করতে পারে।

Discussion questions

১ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং এ প্রতিটি রাউটার তার প্রতিবেশী রাউটারগুলোর সাথে তথ্য বিনিময় করে এবং সেই তথ্য ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি বেলমান-ফোর্ড অ্যালগরিদম ব্যবহার করে কাজ করে।

২ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার পুরো নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ বের করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৩ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

৪ লিঙ্ক স্টেট রাউটিং

লিঙ্ক স্টেট রাউটিং-এ প্রতিটি রাউটার নেটওয়ার্কের একটি সম্পূর্ণ মানচিত্র তৈরি করে এবং ডিজস্ট্রা অ্যালগরিদম ব্যবহার করে সর্বনিম্ন দূরত্বের পথ নির্ধারণ করে। এটি OSPF প্রোটোকলে ব্যবহৃত হয়।

৫ ডিস্টেন্স ভেক্টর রাউটিং

ডিস্টেন্স ভেক্টর রাউটিং-এ রাউটার শুধু তার প্রতিবেশীদের সাথে তথ্য বিনিময় করে। এটি সহজ কিন্তু ধীরগতি এবং লুপ তৈরির ঝুঁকি রয়েছে।

DNS কমপ্লিকেশন থেকে

ডোমেন নেমিং সিস্টেম হল একটি বিতরণকৃত ডেটাবেস যা ডোমেন নেম এবং IP অ্যাড্রেসের মধ্যে মানচিত্র তৈরি করে। এটি ইন্টারনেটের একটি গুরুত্বপূর্ণ উপাদান। এর মাধ্যমে ব্যবহারকারীরা সংখ্যার পরিবর্তে নাম ব্যবহার করে ওয়েবসাইট অ্যাক্সেস করতে পারে।

Optimization

FTP প্রোটোকল ব্যবহার করে ফাইল আদান প্রদান করার জন্য বিভিন্ন পদ্ধতি প্রচলিত আছে। এর মধ্যে একটি হচ্ছে Active FTP। এক্ষেত্রে ক্লায়েন্ট সার্ভারের কাছে কমান্ড পাঠায় এবং সার্ভার ক্লায়েন্টের কাছে ডেটা পাঠায়। কিন্তু ফায়ারওয়াল বা রাউটার থাকলে এটি সমস্যা তৈরি করে। কারণ ফায়ারওয়াল সাধারণত বাইরের কানেকশন ব্লক করে দেয়। এজন্য Passive FTP ব্যবহার করা হয়। এতে ক্লায়েন্ট নিজে সার্ভারের সাথে কানেকশন তৈরি করে এবং ডেটা ট্রান্সফার হয়।

Passive FTP-কেন ব্যবহার করা হয়

Use cases যেমন

ফাইল ট্রান্সফার প্রটোকল (FTP) হচ্ছে একটি স্ট্যান্ডার্ড নেটওয়ার্ক প্রটোকল যা ক্লায়েন্ট-সার্ভার আর্কিটেকচার ব্যবহার করে ফাইল আদান প্রদান করে। এটি TCP/IP নেটওয়ার্ক ব্যবহার করে ডেটা ট্রান্সফার করে। FTP-তে দুইটি চ্যানেল ব্যবহার করা হয়। একটি কমান্ড চ্যানেল এবং অন্যটি ডেটা চ্যানেল। কমান্ড চ্যানেল সাধারণত পোর্ট 21 ব্যবহার করে। ডেটা চ্যানেল ডিফল্ট হিসেবে পোর্ট 20 ব্যবহার করে। এটি সার্ভারের সাথে ক্লায়েন্টের সংযোগ স্থাপন করে। ফাইল ট্রান্সফার, ডিরেক্টরি লিস্টিং এবং অন্যান্য কমান্ড এই চ্যানেলের মাধ্যমে পাঠানো হয়। ডেটা ট্রান্সফার হওয়ার সময় ডেটা চ্যানেল ব্যবহার করা হয়। এটি ফাইল, ডিরেক্টরি লিস্টিং এবং অন্যান্য তথ্য স্থানান্তর করে। FTP-তে দুই ধরনের মোড আছে। Active মোড এবং Passive মোড। Active মোডে সার্ভার ক্লায়েন্টের সাথে সংযোগ স্থাপন করে। Passive মোডে ক্লায়েন্ট সার্ভারের সাথে সংযোগ স্থাপন করে। ফায়ারওয়াল এবং নেটওয়ার্ক কনফিগারেশন অনুযায়ী Passive মোড বেশি নিরাপদ।

প্রটোকল স্কেলিং

প্রত্যেক প্যাকেট একটি নির্দিষ্ট পথ অনুসরণ করে। রাউটারগুলো
সিদ্ধান্ত নেয় কোন প্যাকেট কোথায় যাবে, কিন্তু তারা কখনো
প্যাকেটের ভেতরের তথ্য দেখে না। এটাকে বলা হয়
স্টেটলেস প্রোটোকল, যেটা নেটওয়ার্ক স্কেলিংয়ের জন্য
অত্যন্ত গুরুত্বপূর্ণ।

ডাটা স্কেলিং

আমরা যখন বলি ডাটাবেজ স্কেল করছে, তখন সাধারণত
দুটো জিনিস বোঝায়। একটা হলো ডাটার পরিমাণ বাড়া,
আরেকটা হলো একই সময়ে কতজন ইউজার ডাটা ব্যবহার
করছে। দুটো একই জিনিস নয়, আর দুটোর সমাধান
আলাদা।

ডাটাবেজ স্কেলিং

ডাটাবেজ স্কেলিং মানে হচ্ছে ডাটাবেজ যাতে বেশি ডাটা
সামলাতে পারে এবং বেশি ইউজার একসাথে কাজ করতে
পারে, সেটা নিশ্চিত করা। এটা করার দুটো প্রধান
উপায় আছে, যেগুলো নিয়ে পরের অংশে বিস্তারিত
আলোচনা করা হবে।

ভার্টিক্যাল স্কেলিং

ভার্টিক্যাল স্কেলিং মানে হচ্ছে একটা সার্ভারকে আরও
শক্তিশালী করা। বেশি RAM, দ্রুতগতির CPU, আরও
দ্রুত স্টোরেজ। এটা সহজ, কারণ কোডে কোনো
পরিবর্তন লাগে না। কিন্তু এর একটা সীমা আছে,
এবং খরচ দ্রুত বাড়তে থাকে।

রিপ্লিকেশন

রিপ্লিকেশন মানে হচ্ছে একটা ডাটাবেজের কপি
অন্য সার্ভারে রাখা। একটা প্রাইমারি সার্ভার
লেখার কাজ করে, আর কয়েকটা রিপ্লিকা সার্ভার
পড়ার কাজ করে। যেহেতু বেশিরভাগ অ্যাপ্লিকেশনে
পড়া লেখার চেয়ে অনেক বেশি হয়, এটা খুব
কার্যকর।

শার্ডিং

শার্ডিং মানে হচ্ছে ডাটাবেজকে ছোট ছোট অংশে
ভাগ করা, যেগুলো আলাদা সার্ভারে থাকে। প্রতিটা
অংশকে বলা হয় একটা শার্ড। এটা করা হয় যখন
একটা সার্ভারের ধারণক্ষমতা আর যথেষ্ট নয়।
শার্ডিং খুব শক্তিশালী, কিন্তু এটা অনেক জটিল।

ক্যাশিং

ক্যাশিং মানে হচ্ছে দ্রুত অ্যাক্সেসের জন্য ডাটার
একটা কপি মেমরিতে রাখা। ডাটাবেজ থেকে পড়ার
চেয়ে মেমরি থেকে পড়া অনেক দ্রুত। ক্যাশিং
ডাটাবেজের লোড অনেক কমিয়ে দেয়।

লোড ব্যালান্সিং

লোড ব্যালান্সার হচ্ছে একটা ডিভাইস বা সফটওয়্যার
যেটা ইনকামিং রিকোয়েস্টগুলো কয়েকটা সার্ভারের
মধ্যে ভাগ করে দেয়। ক্লায়েন্ট জানে না কোন
সার্ভার তার রিকোয়েস্ট প্রসেস করছে।

ডাটাবেজ স্কেলিং

ডাটাবেজ স্কেলিং হচ্ছে ডাটাবেজ সিস্টেমকে বড় করা
যাতে সেটা বেশি ডাটা সামলাতে পারে এবং বেশি
ইউজার একসাথে কাজ করতে পারে। এটা করার
দুটো প্রধান উপায় আছে, যেগুলো নিচে আলোচনা
করা হবে।

Xmas Exam

প্রশ্ন -১ ডেফিনিশন পয়েন্ট-ভিত্তিক প্রোগ্রামিং

উত্তর: নড এলগরিদম হচ্ছে একটি সহজ পুনরাবৃত্তিমূলক পদ্ধতি যা কোনো সমস্যার সমাধান খুঁজে বের করে।

Algorithm methods-finding

গণিতে এর অনেক ব্যবহার রয়েছে। উদাহরণ হিসেবে বলা যায় দুটি সংখ্যার গ.সা.গু বের করা।

Euclid's algorithm

গণিতে এর অনেক ব্যবহার রয়েছে। উদাহরণ হিসেবে বলা যায় দুটি সংখ্যার গ.সা.গু বের করা।

প্রশ্ন ২: ফিবোনাচি সংখ্যা কী? উদাহরণসহ ব্যাখ্যা কর।

উত্তর: ফিবোনাচি সংখ্যা হচ্ছে একটি ধারা যেখানে প্রথম দুটি সংখ্যা ০ এবং ১ এবং পরবর্তী সংখ্যাগুলো হচ্ছে আগের দুটি সংখ্যার যোগফল।

Fibonacci sequence

এটি ১৭৭০ সালে গণিতবিদ লিওনার্ডো ফিবোনাচি আবিষ্কার করেন। এটি প্রকৃতিতে অনেক জায়গায় দেখা যায়।

উদাহরণ হিসেবে বলা যায় ফুলের পাপড়ি, গাছের শাখা, শামুকের খোলস ইত্যাদি।

Fibonacci sequence

Silent scan

שולחים גם באקראיות. יודעים שיש IDS וצריך לברר איך הוא מוגדר.
ומחכה x זמן כדי שה IDS לא יגלה קשר בין שליחה לשליחה.

fragmentation scan

קח את החבילות שגדולות איפי ותחתוך אותם

decoy scan

עושה את אותה סריקה אבל מכמה כתובות.
אם שולחים מ-4 מחשבים יש 4 חשודים, פחות סיכוי שיעלו עליי.
אני צריך להיות ברשימה כדי לקבל תשובה.

advanced idle scan

כל חבילה שנשלחת תבוא מכתובת IP שלי.
מה עושה כדי לעקוף איזה פורט פתוח אבל שלא יבוא שזה בא ממני?
1. השתלטות על מחשב אחר שישאל ולקבל את התשובה - פרק השה.
2. משתמשים בזמן שלישי. בכל חבילה יש ID שהוא מס' רץ. מחפשים זומבי,
מישהו שנוכל לנצל אותו ושהוא ללא תנועות IP. מחפש מישהו רדום, מחפשים ללא
מחשבים ללא תנועות IP. נשלח synACK, יחזיר reset ונשמר לו ID כמה פעמים
ונשאל את כל מס' כי הוא מדבר רק איתי. נשלח חבילה מזויפת עם מקור
של הזומבי, הזומבי יחזיר reset, נשלח שוב לזומבי synACK ונקבל reset ונשאל
שקפצנו מס' וזה אומר שהפורט חי.

UDP scan

שולחים לפורט פתוח, הרשת מאזין והוא צריך לענות
שולחים לפורט סגור, הרשת זורק את החבילה ושולח שאי אפשר להגיע אליו
ויודעים כך שהפורט סגור. בכל מחשב אחר נסיק שהפורט פתוח.

port scan prevention

הגדרה של מערכת הפעלה, מחליטים מה מתקינים ומה לא, שירותים שלא רוצים
במחשב להסיר.

OS detection :

নিজে চালাতে হবে। হাতে কলমে করে শিখতে হবে। তাহলে ভুল ধরা যাবে।

Stack Fingerprinting

এটা হচ্ছে TCP/IP স্ট্যাক ইমপ্লিমেন্টেশন। এটা নিয়ে অনেক কাজ করা হয়েছে।

-O nmap -sV -A -vvv -oN -oX -oG -oA

আউটপুট ফরম্যাট

HTTP -iL -iR -exclude

নেটওয়ার্ক রেঞ্জ

আউটপুট ফাইল

Passive : নেটওয়ার্ক ট্রাফিক না পাঠিয়ে

নিজের কম্পিউটারে বসে থেকে নেটওয়ার্ক ট্রাফিক দেখে অন্য কম্পিউটার সম্পর্কে ধারণা করা।

Active : নেটওয়ার্কে ট্রাফিক পাঠিয়ে অন্য কম্পিউটার সম্পর্কে ধারণা করা।

Passive OS detection :

ধরুন আপনি একটা ওয়েব সার্ভার চালাচ্ছেন। আপনার ওয়েব সার্ভারে কেউ একটা ওয়েব পেজ রিকোয়েস্ট পাঠাল। তখন আপনার ওয়েব সার্ভার ওই রিকোয়েস্টের উত্তর দিবে। তখন আপনার ওয়েব সার্ভার ওই রিকোয়েস্টের উত্তর দিবে। তখন আপনার ওয়েব সার্ভার ওই রিকোয়েস্টের উত্তর দিবে। তখন আপনার ওয়েব সার্ভার ওই রিকোয়েস্টের উত্তর দিবে।

ঐ সব লোকের ওপরে
নিজের ইচ্ছে মতো লোকের ওপরে
চলতে চেয়ে।
লোকে লোকে- সরকারি ও সব ওপরে উঠতে গিয়ে ও সাফল্যের জন্য।

Fight for status

হিংসা মারামারি গিয়ে স্বার্থের জন্য। অথবা যখন বেশি বেশি লোকে ধর্ম
লোকের হানির বা লোকের- Ego-Ego-Clash
Role-Conflict-Problem ও সামাজিক
0-Year - System Part
Part Number

লোকের যা, সবে মিলে একটা সমাজ।
এক রাজা গেলেন বনে, সেই বনে আরো রাজা এসে
যা পাবে পাবেন, মানিয়ে, নিজের মতো করে নিবেন।

এই লেখাতে ও নতুন কুসংস্কার (ডেভিলস, ভূতের ভয়)
হচ্ছে যেগুলি লোকে কুসংস্কারে
বলছে, যেমন 2:

নতুনভাবে Security-Concern

12.4.24

১.৮.১৪ সেমিকন্ডাক্টর

ট্রান্সিস্টর

ট্রান্সিস্টর হচ্ছে সেমিকন্ডাক্টর দিয়ে তৈরি একটি ডিভাইস যা ভোল্টেজ বা কারেন্ট নিয়ন্ত্রণ করে। এটি সাধারণত তিনটি টার্মিনাল বিশিষ্ট হয়। ট্রান্সিস্টর সুইচ হিসেবে এবং অ্যামপ্লিফায়ার হিসেবে কাজ করে। এটি ডিজিটাল ও অ্যানালগ সার্কিটের মূল উপাদান।

ট্রান্সিস্টর

ট্রান্সিস্টর-এর প্রকারভেদ

বাইপোলার জাংশন ট্রান্সিস্টর

ফিল্ড ইফেক্ট ট্রান্সিস্টর

ট্রান্সিস্টর হচ্ছে সেমিকন্ডাক্টর দিয়ে তৈরি একটি ডিভাইস যা ভোল্টেজ বা কারেন্ট নিয়ন্ত্রণ করে। এটি সুইচ হিসেবে এবং অ্যামপ্লিফায়ার হিসেবে কাজ করে। ট্রান্সিস্টর তিনটি টার্মিনাল বিশিষ্ট হয়। ট্রান্সিস্টর ডিজিটাল ও অ্যানালগ সার্কিটের মূল উপাদান।

ফিল্ড ইফেক্ট ট্রান্সিস্টর

ফিল্ড ইফেক্ট ট্রান্সিস্টর হচ্ছে একটি সেমিকন্ডাক্টর ডিভাইস যা বৈদ্যুতিক ক্ষেত্র ব্যবহার করে কারেন্ট নিয়ন্ত্রণ করে। এটি ভোল্টেজ-নিয়ন্ত্রিত ডিভাইস। এর উচ্চ ইনপুট ইমপেডেন্স রয়েছে এবং এটি কম শক্তি ব্যবহার করে। এটি ডিজিটাল ও অ্যানালগ সার্কিটে ব্যবহৃত হয়।

ফিল্ড ইফেক্ট ট্রান্সিস্টর - জংশন ফিল্ড ইফেক্ট ট্রান্সিস্টর

মেটাল - অক্সাইড - সেমিকন্ডাক্টর ফিল্ড ইফেক্ট ট্রান্সিস্টর

JFET-এর গঠন ও কার্যপ্রণালী

JFET-এর গঠন হচ্ছে একটি n-টাইপ বা p-টাইপ সেমিকন্ডাক্টর চ্যানেল যার দুই প্রান্তে দুটি টার্মিনাল থাকে। এটি ভোল্টেজ-নিয়ন্ত্রিত ডিভাইস। গেট-সোর্স জংশন রিভার্স বায়াস করা হয়। ড্রেন-সোর্স ভোল্টেজ প্রয়োগ করলে ড্রেন কারেন্ট প্রবাহিত হয়। গেট ভোল্টেজ পরিবর্তন করে চ্যানেলের প্রস্থ নিয়ন্ত্রণ করা হয়।

JFET-এর গঠন

JFET-এর গঠন হচ্ছে একটি n-টাইপ বা p-টাইপ সেমিকন্ডাক্টর চ্যানেল যার দুই প্রান্তে দুটি টার্মিনাল থাকে। এটি ভোল্টেজ-নিয়ন্ত্রিত ডিভাইস। এর উচ্চ ইনপুট ইমপেডেন্স রয়েছে এবং এটি কম শক্তি ব্যবহার করে। এটি ডিজিটাল ও অ্যানালগ সার্কিটে ব্যবহৃত হয়।

JFET-এর কার্যপ্রণালী

JFET-এর কার্যপ্রণালী ড্রেন-সোর্স ভোল্টেজ প্রয়োগ করলে ড্রেন কারেন্ট প্রবাহিত হয়। গেট ভোল্টেজ পরিবর্তন করে চ্যানেলের প্রস্থ নিয়ন্ত্রণ করা হয়। গেট-সোর্স জংশন রিভার্স বায়াস করা হয়। ড্রেন-সোর্স ভোল্টেজ প্রয়োগ করলে ড্রেন কারেন্ট প্রবাহিত হয়। গেট ভোল্টেজ পরিবর্তন করে চ্যানেলের প্রস্থ নিয়ন্ত্রণ করা হয়।

JFET-এর গঠন

JFET-এর গঠন হচ্ছে একটি n-টাইপ বা p-টাইপ সেমিকন্ডাক্টর চ্যানেল যার দুই প্রান্তে দুটি টার্মিনাল থাকে। এটি ভোল্টেজ-নিয়ন্ত্রিত ডিভাইস। এর উচ্চ ইনপুট ইমপেডেন্স রয়েছে এবং এটি কম শক্তি ব্যবহার করে। এটি ডিজিটাল ও অ্যানালগ সার্কিটে ব্যবহৃত হয়।

JFET-এর কার্যপ্রণালী

JFET-এর কার্যপ্রণালী ড্রেন-সোর্স ভোল্টেজ প্রয়োগ করলে ড্রেন কারেন্ট প্রবাহিত হয়। গেট ভোল্টেজ পরিবর্তন করে চ্যানেলের প্রস্থ নিয়ন্ত্রণ করা হয়। গেট-সোর্স জংশন রিভার্স বায়াস করা হয়। ড্রেন-সোর্স ভোল্টেজ প্রয়োগ করলে ড্রেন কারেন্ট প্রবাহিত হয়। গেট ভোল্টেজ পরিবর্তন করে চ্যানেলের প্রস্থ নিয়ন্ত্রণ করা হয়।

ב black box, זה שאנחנו מחלקים הכל לאלגוריתם ומפתח ואת הקרא.

type of operation:

substitution - החלפת כל אות במשהו אחר, באות, בנקודה.

transposition - החלפת המיקום של האותיות

מנגנים לצפנה

plain text הטקסט הרגיל

cipher text הטקסט אחרי הצפנה

encryption algorithm - האלגוריתם לצפנה

(S)עשה - סימטרי או א-סימטרי

cipher text - הטקסט המוצפן (מה לבד המפתח)

decryption algorithm - האלגוריתם לפענח

Kerckhoff's principle

העקרון - מצפינים מול A, B רוצה לפענח. כל הנתונים לגבי האלגוריתם, המפתח ומי B יפענח אבל לא יכול את המפתח. בלי המפתח א' אפשר עשות כלום.

work factor - כמה אני צריך להשקיע כדי לפענח את המפתח.
כולם שם work factor יהיה גבוה, שהוא יהיה יותר גבוה מהזמן שרמיקל שווה. למשל, שיהיה שווה מליון שנה ומיליון סה אלף.
או שיהיה יותר כמו מהזמן שהמיקל כלומר.

cryptanalysis

brute force - נעבור את כל האפשרויות

exploit weakness - חיפוש חולשות באלגוריתם הצפנה.

brute force:

אורך מפתחות בביט: 32, 56, 128, 168

ב-32 ביט לוקח 32 דקות למצוא את כל האפשרויות. [באמצע בחיים]

ב 168 ביט לוקח $5.9 \cdot 10^{36}$ שנים.

במפתח עם ביט גבוה אין מה לדבר על brute force, זה יקח יותר מידי זמן.

ফলে ক্ষেত্রে দাঁড় করাতে হবে যাতে তা সঠিক তথ্য দেয়।

Polyalphabet: সুইচিংকৃত

পরিচয় ও প্রমাণ ছাড়া আমার তা দাঁড়ায়

প্রতিটি ক্যারেক্টারের জন্য

$C_i = E_K(P_i) = (P_i + K) \bmod 26$

ডিক্রিপ্ট: $P_i = D_K(C_i) = (C_i - K) \bmod 26$

$K = 3$ ফরমুলা

Ceasar cipher 26 ও :ডিক্রিপ্ট

Monoalphabetic: সুইচিংকৃত

কীবোর্ডের 26 বর্ণ এবং 26 বর্ণ মিলে 26! ফ্যাক্টরিয়াল বা 26! কম্বিনেশন হয়। এটা ব্রুট ফোর্স করে ভাঙা অসম্ভব। কারণ এতে ভেঙে দেয়ার জন্য প্রতিটি ক্যারেক্টারের ফ্রিকোয়েন্সি বিশ্লেষণ করতে হয়।

Cryptanalysis: ক্রিপটো-এনালিসিস

ক্রিপটোগ্রাফি ভাঙার জন্য ক্রিপটো-এনালিসিস করতে হয়। এতে প্লেইনটেক্সট ও সাইফারটেক্সট উভয়ই জানা থাকে। এর মধ্যে Known plain text, Chosen plaintext ও Chosen ciphertext অন্যতম।

Type of attacks

প্লেইনটেক্সট ও সাইফারটেক্সট জানা থাকা। 2. Known plain text: প্লেইনটেক্সট ও সাইফারটেক্সট জানা থাকা। 3. Chosen plaintext: চয়েজ প্লেইনটেক্সট দিয়ে সাইফারটেক্সট বের করা।

১৭ তম ৫- ৭ তম ঘণ্টা

ঘণ্টা ১১০ ০১০০

ঘণ্টা ১১০ ০১০০ তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো ব্যবহার করে ৩০ তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো দিয়ে পরীক্ষা করা যায় কি না। না যায় কেন? ব্যাখ্যা কর।

উত্তর: না যায় না। কারণ ১১০ তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো ৩০ তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো অপেক্ষা বড়। বড় তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো ছোট তরঙ্গদৈর্ঘ্য বিশিষ্ট বস্তু দ্বারা বিক্ষেপিত হয় না। তাই ১১০ তরঙ্গদৈর্ঘ্য বিশিষ্ট আলো ৩০ তরঙ্গদৈর্ঘ্য বিশিষ্ট বস্তু দ্বারা বিক্ষেপিত হবে না। ফলে ৩০ তরঙ্গদৈর্ঘ্য বিশিষ্ট বস্তু পর্যবেক্ষণ করা যাবে না।

গাণিতিক সমস্যা:

৭৫ ওয়াট = ৭৫ জুল

বাল্বের প্রতিদিনের ব্যবহৃত শক্তি

বিদ্যুৎ বিল কত?

ব্যয়িত শক্তি = ৭৫ জুল × ৪ ঘণ্টা × ৩০ দিন = ৯০০০০০ জুল

= ৯০০০০০ / ১০০০ কিলোজুল

= ৯০০ কিলোজুল

= ৯০০ / ৩.৬ কিলোওয়াট ঘণ্টা

= ২৫০ কিলোওয়াট ঘণ্টা

বিদ্যুৎ বিল = ২৫০ × ৫ টাকা = ১২৫০ টাকা

ব্যাখ্যা: বিদ্যুৎ বিল নির্ণয়ের জন্য প্রথমে ব্যয়িত শক্তি বের করতে হবে। ব্যয়িত শক্তি বের করার জন্য বাল্বের ক্ষমতা এবং ব্যবহার করা সময় গুণ করতে হবে। এরপর কিলোওয়াট ঘণ্টায় রূপান্তর করে বিদ্যুৎ বিল নির্ণয় করা যায়।

সৃজনশীল প্রশ্ন:

১। বিদ্যুৎ বিল কোথায়?

উত্তর: বিদ্যুৎ বিল ঢাকায়।

২। বিদ্যুৎ বিল কত?

উত্তর: বিদ্যুৎ বিল ১২৫০ টাকা।

৩। বিদ্যুৎ বিল নির্ণয়ের জন্য কি কি প্রয়োজন?

উত্তর: বিদ্যুৎ বিল নির্ণয়ের জন্য ক্ষমতা, ব্যবহার করা সময় এবং একক মূল্য প্রয়োজন।

৪। বিদ্যুৎ বিল কিভাবে নির্ণয় করা হয়?

উত্তর: বিদ্যুৎ বিল নির্ণয় করা হয় ব্যয়িত শক্তি বের করে এবং তা কিলোওয়াট ঘণ্টায় রূপান্তর করে।

৫। বিদ্যুৎ বিল কত হবে?

উত্তর: বিদ্যুৎ বিল ১২৫০ টাকা হবে।

৬। বিদ্যুৎ বিল কেন বেশি?

উত্তর: বিদ্যুৎ বিল বেশি কারণ বাল্বের ক্ষমতা বেশি এবং ব্যবহার করা সময় বেশি।

৭। বিদ্যুৎ বিল কেন কম?

উত্তর: বিদ্যুৎ বিল কম কারণ বাল্বের ক্ষমতা কম এবং ব্যবহার করা সময় কম।

৮। বিদ্যুৎ বিল কিভাবে কমানো যায়?

উত্তর: বিদ্যুৎ বিল কমানো যায় কম ক্ষমতার বাল্ব ব্যবহার করে এবং কম সময় ব্যবহার করে।

৯। বিদ্যুৎ বিল কিভাবে বাড়ানো যায়?

উত্তর: বিদ্যুৎ বিল বাড়ানো যায় বেশি ক্ষমতার বাল্ব ব্যবহার করে এবং বেশি সময় ব্যবহার করে।

১০। বিদ্যুৎ বিল কিভাবে নির্ণয় করা হয়?

উত্তর: বিদ্যুৎ বিল নির্ণয় করা হয় ব্যয়িত শক্তি বের করে এবং তা কিলোওয়াট ঘণ্টায় রূপান্তর করে।

১১। বিদ্যুৎ বিল কত হবে?

উত্তর: বিদ্যুৎ বিল ১২৫০ টাকা হবে।

১২। বিদ্যুৎ বিল কেন বেশি?

উত্তর: বিদ্যুৎ বিল বেশি কারণ বাল্বের ক্ষমতা বেশি এবং ব্যবহার করা সময় বেশি।

১৩। বিদ্যুৎ বিল কেন কম?

উত্তর: বিদ্যুৎ বিল কম কারণ বাল্বের ক্ষমতা কম এবং ব্যবহার করা সময় কম।

১৪। বিদ্যুৎ বিল কিভাবে কমানো যায়?

উত্তর: বিদ্যুৎ বিল কমানো যায় কম ক্ষমতার বাল্ব ব্যবহার করে এবং কম সময় ব্যবহার করে।

১৫। বিদ্যুৎ বিল কিভাবে বাড়ানো যায়?

উত্তর: বিদ্যুৎ বিল বাড়ানো যায় বেশি ক্ষমতার বাল্ব ব্যবহার করে এবং বেশি সময় ব্যবহার করে।

הצפנה סימטרית

האלגוריתם של הצפנה ופיענוח הם פונקציה.
המפתח צריך להיות סודי.
המפתח הוא משותף כי צריך להיות לשני הצדדים.
מצפינים בעזרת מפתח אחד ויש מפענחים בעזרתו

הצפנה: $C = E_K(P)$

פיענוח: $P = D_K(C)$

modern encryption:

XOR:

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

XOR בין שני סמלים תמיד 0.
XOR עם 0 אי אפשר את סמל.
XOR עם B ואז שוב עם B אין
הוצר סמל.
A- plain text
B- מפתח.

XOR properties:

$A \oplus A = 0$

$A \oplus 0 = A$

$(A \oplus B) \oplus B = A$

הבעיות בהצפנה סימטרית זה הלבנת המפתח.

Electronic code book (ECB):

לוקח את הודעה, אולם מחשבות בלוקים (בלוקים) ולא חשבה מצפינים.
כל בלוק - 64 ביט.
הבעיה- יש pattern שנשאר בהצפנה.
פתרון- לבלבן את הסיפר או לחסל משהו שונה.

cipher block chaining (CBC)

אם לחלק לבלוקים אבל כל בלוק לעשות מין XOR בינם.
עושים XOR ואז מצפינים.
כדי לפענח עושים decrypt ואז XOR.
פה השגיאה לוקחת, כ ECB לא ואפשר לעשות מקבל.

ধরি f একটি

DES - এর মূল প্রস্তাবিত এ-এনক্রিপশন

অ্যালগরিদম ছিল।

এতে ডেটাকে প্রতিটি ৬৪ বিটের ব্লকে ভাগ করা হতো। প্রতিটি ব্লক আবার ৩২ বিটের দুটি অংশে ভাগ করা হতো। এরপর ডানের অংশটি ৪৮ বিটে এক্সপ্যান্ড করা হতো। এরপর সেটি ৫৬ বিটের একটি সাবকি-র সাথে XOR করা হতো। এরপর ফলাফলকে ৮ টি ৬ বিটের ব্লকে ভাগ করা হতো। প্রতিটি ব্লক একটি নির্দিষ্ট S-box দিয়ে প্রক্রিয়া করা হতো। এরপর ফলাফলকে আবার ৩২ বিটে কমানো হতো। এরপর ফলাফলকে বাম অংশের সাথে XOR করা হতো। এরপর বাম অংশ ও ডান অংশ পরস্পর বদলে যেত। এই প্রক্রিয়া ১৬ বার পুনরাবৃত্তি হতো। শেষে বাম ও ডান অংশ একত্রে যুক্ত হয়ে ৬৪ বিটের ব্লক তৈরি হতো। এরপর ফলাফলকে আবার প্রারম্ভিক ক্রমচয়ন দিয়ে পরিবর্তন করা হতো। এটিই হলো DES (DES) এর কাজের পদ্ধতি।

AES

AES এর পূর্ণরূপ হলো Advanced Encryption Standard। এটি একটি প্রতীকী এনক্রিপশন অ্যালগরিদম। এটি ব্লক সাইফার হিসেবে কাজ করে। এটি ১৯৯৭ সালে NIST কর্তৃক প্রস্তাবিত হয়। এটি ১২৮ বিট ব্লক ব্যবহার করে। এটি ১২৮, ১৯২ অথবা ২৫৬ বিট কী ব্যবহার করে। এটি ১০, ১২ অথবা ১৪ রাউন্ড চালায়। এটি ৪ বাইটের ব্লকে কাজ করে। এটি ৪ বাইটের ব্লকে কাজ করে। এটি ৪ বাইটের ব্লকে কাজ করে। এটি ৪ বাইটের ব্লকে কাজ করে।

(E) সূত্র: $C = Enc(D_{k_2}(Enc(P)))$

(D) সূত্র: $P = D_{k_1}(Enc(D_{k_2}(C)))$

Location of maximum deviation

প্রথম ধাপ হবে ‘b’ কোন বিন্দুতে বক্র এবং x অক্ষের মধ্যে সর্বোচ্চ দূরত্ব হয়। এটা বের করার জন্য আমাদের x এর সাপেক্ষে y' কে অন্তরকলন করতে হবে।

$$y' = \frac{1}{2}x - \frac{1}{2}$$

এখন y' কে শূন্য সমান সেট করে পাই $x = 1$ এটা হচ্ছে সেই বিন্দু যেখানে বক্র এবং x অক্ষের মধ্যে সর্বোচ্চ দূরত্ব আছে। এখন $x = 1$ কে মূল সমীকরণে বসিয়ে পাই $y = -\frac{1}{4}$ এটা হচ্ছে সেই বিন্দু যেখানে বক্র এবং x অক্ষের মধ্যে সর্বোচ্চ দূরত্ব আছে।

এখন আমরা দেখব যে বক্রটি কোন বিন্দুতে x অক্ষকে ছেদ করে। এটা বের করার জন্য আমরা $y = 0$ বসাব এবং x এর জন্য সমাধান করব। $x = 0$ এবং $x = 2$ এটা হচ্ছে সেই বিন্দু যেখানে বক্রটি x অক্ষকে ছেদ করে।

Location of maximum deviation

প্রথম ধাপ হবে ‘b’ কোন বিন্দুতে বক্র এবং x অক্ষের মধ্যে সর্বোচ্চ দূরত্ব আছে। এটা বের করার জন্য আমাদের x এর সাপেক্ষে y' কে অন্তরকলন করতে হবে।

$$y' = \frac{1}{2}x - \frac{1}{2}$$

পুয়াসঁ মডেল

মডিউলেশন

ধরি, একটি র‍্যান্ডম ভেরিয়েবল X-এর ডেনসিটি ফাংশন $f(x)$ এবং এর মোমেন্ট জেনারেটিং ফাংশন $M_X(t)$। তাহলে X-এর ডিস্ট্রিবিউশন $f(x)$ এর সাথে $M_X(t)$ এর একটি ওয়ান টু ওয়ান ম্যাপিং আছে। অর্থাৎ $M_X(t)$ জানা থাকলে $f(x)$ বের করা যায়।

ধরি, X একটি র‍্যান্ডম ভেরিয়েবল যার ডেনসিটি ফাংশন $f(x)$ এবং মোমেন্ট জেনারেটিং ফাংশন $M_X(t)$। আবার Y একটি র‍্যান্ডম ভেরিয়েবল যার ডেনসিটি ফাংশন $g(y)$ এবং মোমেন্ট জেনারেটিং ফাংশন $M_Y(t)$। যদি $M_X(t) = M_Y(t)$ হয়, তাহলে $f(x) = g(y)$ হবে।

অর্থাৎ X এবং Y একই ডিস্ট্রিবিউশন অনুসরণ করবে।

ধরি, X একটি র‍্যান্ডম ভেরিয়েবল যার ডেনসিটি ফাংশন $f(x)$ এবং মোমেন্ট জেনারেটিং ফাংশন $M_X(t)$। তাহলে X-এর ডিস্ট্রিবিউশন $f(x)$ এর সাথে $M_X(t)$ এর একটি ওয়ান টু ওয়ান ম্যাপিং আছে। অর্থাৎ $M_X(t)$ জানা থাকলে $f(x)$ বের করা যায়।

ধরি, X একটি র‍্যান্ডম ভেরিয়েবল যার ডেনসিটি ফাংশন $f(x)$ এবং মোমেন্ট জেনারেটিং ফাংশন $M_X(t)$। আবার Y একটি র‍্যান্ডম ভেরিয়েবল যার ডেনসিটি ফাংশন $g(y)$ এবং মোমেন্ট জেনারেটিং ফাংশন $M_Y(t)$। যদি $M_X(t) = M_Y(t)$ হয়, তাহলে $f(x) = g(y)$ হবে।

অর্থাৎ X এবং Y একই ডিস্ট্রিবিউশন অনুসরণ করবে।

Sequence model

$M_X = M_Y$:

অর্থাৎ $f(x) = E_X[e^{tX}]$ $\quad$ $m = \text{পজিটিভ}$

ডেফিনিশন $M_X = D_{tX}(c) = D_{tX}[E_X[e^{tX}]]$ $\quad$ $c = \text{কোনো ফিক্সড}$

Autoregressive model

সিগনাল : $c = E_{tX}(x)$

রেগ্রেশন : $M = D_{tX}(c) = D_{tX}[E_X[e^{tX}]]$

Security of RSA cryptosystem model. Combined

RSA

এটি একটি অসিমেট্রিক এলগরিদম যা দুটি বড় প্রাইম নাম্বার ব্যবহার করে তৈরি করা হয়।

এটি একটি পাবলিক কি ক্রিপ্টোগ্রাফি। এতে দুটি কি ব্যবহার করা হয়। একটি পাবলিক কি এবং অন্যটি প্রাইভেট কি। পাবলিক কি দিয়ে এনক্রিপ্ট করা মেসেজ শুধু প্রাইভেট কি দিয়ে ডিক্রিপ্ট করা যায়।

Key exchange

এটি একটি সিমেট্রিক কি এলগরিদম যেখানে দুজন ব্যক্তি একই কি ব্যবহার করে মেসেজ এনক্রিপ্ট করে এবং ডিক্রিপ্ট করে।

$c = m^e \bmod n$

$m = c^d \bmod n$

$m = (\underbrace{m^e \bmod n}_{c})^d \bmod n$

$e.d \bmod z = 1 (e.d-1)$ এটি একটি সংখ্যা যা $p-1$ ও $q-1$ এর গুণিতক হয় এবং z এর সাথে কোপ্রাইম হয়।

$n = p.q$ এবং $z = (p-1)(q-1)$

:এখন ধরা যাক: z এর একটি গুণিতক e এবং d এর মধ্যে $e.d$ এর গুণফল z এর সাথে মডিউলাস হিসেবে 1 হয়। z এর সাথে কোপ্রাইম e এর একটি মডিউলার ইনভার্স d আছে।

$e.d \bmod z = 1$

$z = 1 \cdot e \cdot d$ মডিউলার ইনভার্স d এর একটি অনন্য মান আছে যা 1 এবং z এর মধ্যে হবে।

:উদাহরণ: $p = 7$, $q = 11$

$n = 7 \times 11 = 77$

$z = (7-1)(11-1) = 6 \times 10 = 60$

$e = 13$ এবং $d = 37$ হলে $e.d = 13 \times 37 = 481$

$481 \bmod 60 = 1$

$e.d \bmod z = 1$

$e.d \bmod z = 1(e.d-1)$ এটি একটি সংখ্যা যা $p-1$ ও $q-1$ এর গুণিতক হয় এবং z এর সাথে কোপ্রাইম হয়।

$n = p.q$ এবং $z = (p-1)(q-1)$

:এখন ধরা যাক: z এর একটি গুণিতক e এবং d এর মধ্যে $e.d$ এর গুণফল z এর সাথে মডিউলাস হিসেবে 1 হয়। z এর সাথে কোপ্রাইম e এর একটি মডিউলার ইনভার্স d আছে।

$e.d \bmod z = 1$

$z = 1 \cdot e \cdot d$ মডিউলার ইনভার্স d এর একটি অনন্য মান আছে যা 1 এবং z এর মধ্যে হবে।

:উদাহরণ: $p = 7$, $q = 11$

$n = 7 \times 11 = 77$

$z = (7-1)(11-1) = 6 \times 10 = 60$

$e = 13$ এবং $d = 37$ হলে $e.d = 13 \times 37 = 481$

$481 \bmod 60 = 1$

$e.d \bmod z = 1$

এমনটি হওয়াটা 'হতে পারে', নাকি 'হবে' বলা
হতে পারে নাকি নয়তে পারে, যে এমনটি হওয়ার কোন সম্ভাবনা নেই

ঠা.গ. 'হবে'

সম্ভাবনা।

এই- নাকি সম্ভাবনা প্রকাশের জন্য -তে
-তে - নাকি সম্ভাবনা প্রকাশে ব্যবহৃত একটি শব্দ যা সম্ভাবনা অর্থবোধক
হয়ত সম্ভবপর

পানাসেরি হয়

নতুন একটি গণ এনকোড যা হতে পারে সম্ভাবনা

হতে পারে। যা প্রথম নাম-কিছুর পরিবর্তে ?
কখন না ও। যা প্রথম নাম-কিছুর পরিবর্তে ?

Fish পরিবর্তে

প্রত্যেক প্রাণী দুই নামের একটি বৈজ্ঞানিক নাম -টি
প্রথম অংশটি হচ্ছে 'জেনারা' (G) বিতীয় অংশ হচ্ছে 'স্পিশিজ' (s)
এ দুটিই হচ্ছে লাতিন ভাষায় এবং ইটালিক হবে।

সম্ভবত প্রজাতি হচ্ছে এর মধ্যে একটি গণ = কমন ফিচার

3.5.19 RSA-המשך

רוצים לקחת מפתח ציבורי של מישהו ולהיות בטוחים שזה הציבורי שזה של אותו אדם, לוקחים מישהו שסומכים עליו ונקרא לו trusted, הוא יהיה third party. שני הצדדים צריכים לפגוש לפחות פעם אחת של הצד השלישי ולקבל את המפתח הציבורי שלו.

אם יש את הציבורי של הצד השלישי וסומכים עליו, הולכים אליו, נותנים לו לחתום עם הציבורי שלו ורק יודעים שהוא מאמת את האדם שפגש. פוגשים פיזית את הצד השלישי ולכן אי אפשר להתחזות אליו והציבורי בהכרח שלו.

לצד השלישי קוראים CA - certification authority, והוא מחלק תעודות, החתום על ידו נקרא certificate.

ה CA צריך הרבה כסף. הוא לא יכול לפגוש לכל מי שרוצה ולכן הוא בוחר מישהו שהוא סומך עליו שיהיה מוסמך לתת אישורים. ה CA חותם שהוא מכיר אותו ונותן לו תוקף אישור שהוא מוסמך. כך אפשר להסמיך עוד וזה היררכי.

בהיררכיה בודקים אחורה עד שמגיעים ל CA שמכירים וכך מכירים את כל המוסמכים בהיררכיה.

בדפדפן בודקים את המוסמכים, יש כאלה שרשומים ויש כאלה שצריכים לסמוך לבד.

מה יש ב certificate:

תאריך התחלה ותאריך סוף.

message integrity - hash function

hash זו פונקציה חד כיוונית, לוקחת משהו גדול למשהו קטן.

אין דרך הפוכה, אי אפשר לחזור למה שהיה לפני.

משתמשים ל message integrity.

כל צד עושה hash, הצד המקבל עושה hash ומוודא שהתשובה אותו דבר.

בעיה- מישהו בדרך לוקח את המייל, עושה hash חדש ושם אותו.

שומרים על אמינות המידע.

זה טוב אם יש בעיה, תקלה בתקשורת.

לא יעבוד אם מישהו ייקח את ההודעה בדרך ויעשה hash חדש.

check sum - מתבסס ביטים ויוצרת תוצאה שמראים שעם תהיה
תקלת תקשורת יהיו. יהיו שהודעה שונתה מקבל תהלה.
הכל עושות שינוי בהודעה בלי שה hash ישתנה.
מלכתחילה הבעיה שאומת hash של סיסמא ולא את הסיסמא עצמה.
ההודעה יכולה להיות בכל גודל, ה hash יהיה באותו גודל, צריך
לחשב את ה hash בצורה הלה.
נבנה hash ולא נצטרך לחלוא הודעה שצר ה hash שלה.
אותה הודעה- אותו hash.
מציאת 2 הודעות שה hash שלהם לא אותו ערך - אי אפשרי.
check sum - hash לא טובה.

birthday attack

23 אנשים יש סיכוי של 6% של אותו יום הולדת עם איש א'.
23 אנשים יש סיכוי של 50% של 2 אנשים עם אותו תאריך יום
הולדת
כך אפשר לנצל hash פיתר טוב.

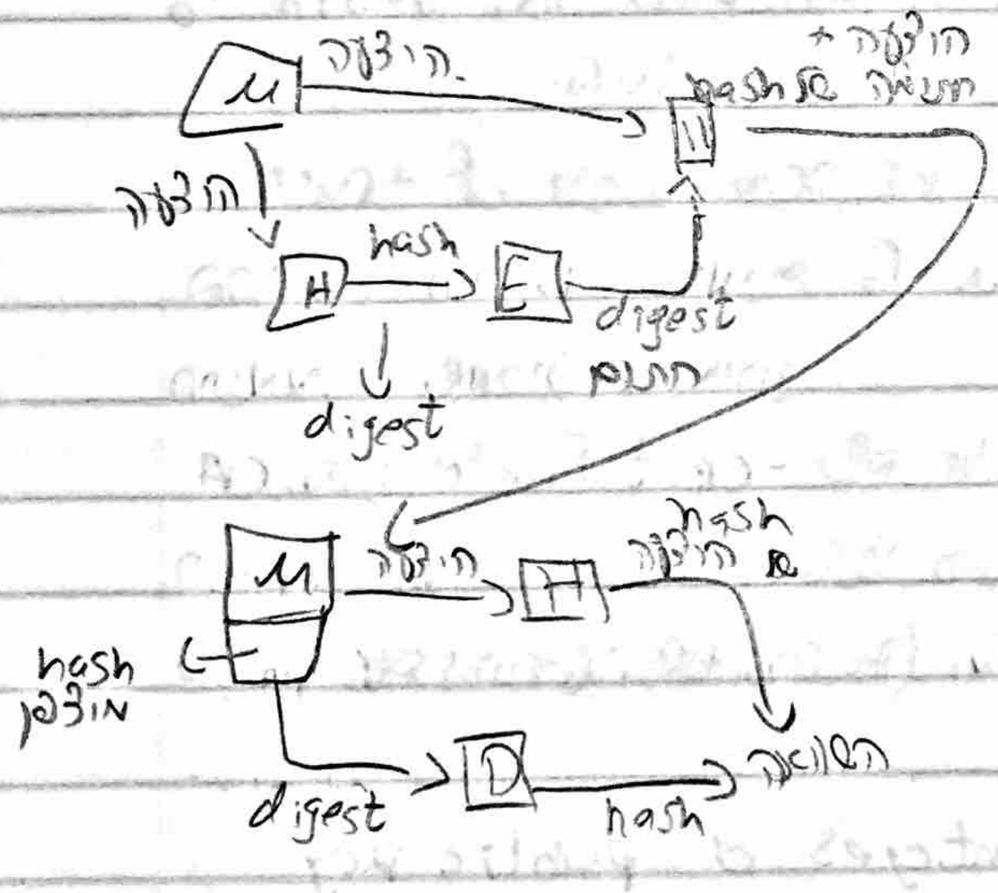
main hash algorithm

MD5 - משתמש בבלוקים של 512 ותוצאת ה hash היא 128.
SHA1 - משתמש בבלוקים של 512 ותוצאת ה hash היא 160.
MD5 כבר לא מומלץ.
SHA1 נפרץ.
SHA256 נמצא גם ב SHA2 ו SHA3-1 ה output שלהם
גדול יותר ויותר קשה לפרוץ.

message authentication code (MAC)
לא MAC כתובת פיזית.
משתמשים בפונקציה ומופיעים מפתח סודי וכך אי אפשר לעשות
man in the middle ושינוי את ההודעה גם hash.

חתימה דיגיטלית

רוצים לחסוך ולעשות את החתימה בסימטרי.
לוקחים את ההודעה הארוכה, עושים hash ומוסיפים רק לה.
במקום להצפין את כל ההודעה עושים רק לה.
לכה הופכים חתימה.
שהיא קטן - יכול להיות במהרה.
לוקחים הודעה, עושים לה hash, את ה hash מצפינים בעזרת הפרטי
שלו, מחברים את זה להודעה.
הלב השני, לוקח את החתימה ועושה סנכרון לחתימה עם הציבורי, מקבלים
את ה hash. הלב השני עושה hash, משווה אותם ורואה שזו אותה
הודעה אם ה hash זהה. עושה hash להודעה ומשווה.
אי אפשר להחליף hash להודעה.
שאלה: נרצה את השטרות, מנתק מה שבפנים ושם מסגרים. למלא את
מה שבפנים (בשטרות).
שאלה: מסמן בשטרות כל מה ושואל מה יש לה כל מה.
תשובה:



hash - digest מוצפן
signed digest - אותו הדבר
message - M
encryption - E
hash - H
decryption - D
חתימה - hash מוצפן

שקף 67:
man in the middle הוא המסוכן ב hash, אין authentication.
התבחשות יש ב hash, mac, אין בחתימה.
אפשרות, אין ב hash, ב mac סימטרי ובחתימה פיזיקלית זו סימטרי.

diffie-hellman בין man in the middle,

public key infrastructure

אם נניח לאישור את הפרטי, ה CA מבטל לו את ה certificate.
לא רשום מליון מבוטל. יש רשימה של revoked, הדפדפן בודק לפניך
שהה certificate לא מבוטל.
אם אישור נתן certificate לכמה אנשים, ושהו מבוטל, כל אלה שנתן
להם גם שלהם מבוטל.
בדפדפן בודקים את האישור שהוא תקין ואל בודקים ברשימת מבוטלים.
EV - extented validation. בדיקה יותר משמעותית, יתכן יהיה. בנקים,
תבנות סחר אלקטרוני (איבי, אמזון) מחייבים.
certificate - X.509.
chain of trust - היררכיה של CA. בדרך 3 רמות יכולים.
הכם בנתונים מי לפניך לא פונים לשל אחד. כשמשתמש מדפדף איתו הוא
פונה אלי ונתן לי כבר חתימה ומנשקים עם החתימה שלו נגיות. אימת
לי הדפדפן, זה רשום שם, לא פונים לשל אחד.
top down - מלמעלה למטה.
web of trust - אישות לא מאושר הלכה ושולחים עם מכירים אותו
ברשת. תלוי כמה סומכים על אישות. הלא מאושר יכול מי לפניך, יתן
חתימות שמכירים אותו
CA מונה certificate - אפס או עסק ל CA ואומר שמדוע רוצה איש,
הוא נותן ל CA את הלבנשי, מושה hash (ה CA), ולוקח digest, CA
מצפין עם הפרטי שלו (חתימה), מגדפים את החתימה לבנוי ל אפס א'.

advanteges of public key

מאוד שמוש בסביבה פתוחה.
מפתחות הצפנה הם גבשים אבל עובדא שהפרטי אכן של מי שרוצים
לפגוש איתו.

disadvanteges of public key

שאלה: הצפנה א-סימטרית יכולה יקרה מהצפנה סימטרית. איך פעשים
את הבעיה?
תשובה: עדיר את המפתח ב א-סימטרי ושמר בסימטרי. מתרגם מהצפן

বে সাত ও পুরুষ'

১। বিবাহের : বাবার বাড়ি থেকে স্বামীর বাড়িতে যাওয়া। (সাধারণ) ১৫ বছর বয়সে বিবাহ হয়ে শ্বশুরবাড়ি চলে যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়িতে যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়ি যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়ি যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়ি যাওয়া।

২। বিবাহের পর বাড়ি থেকে স্বামীর বাড়ি যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়ি যাওয়া। বাবার বাড়ি থেকে স্বামীর বাড়ি যাওয়া।

বিবাহের পর বাড়ি থেকে স্বামীর বাড়ি যাওয়া।

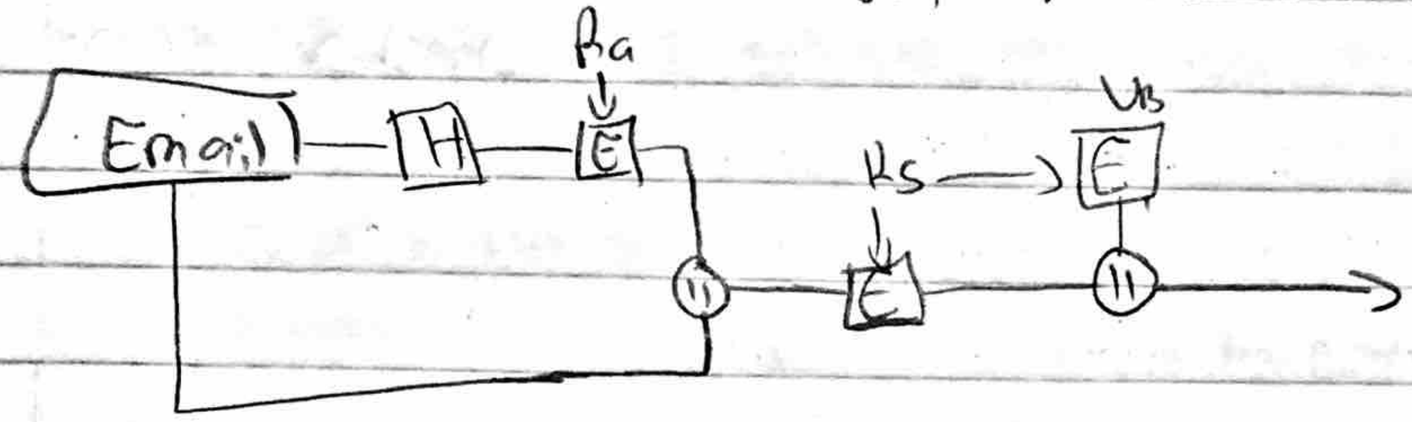
Electronic mail security

הצפנת מיילים

PGP:

איך שולחים מיילים מאובטחים?

בהצפנה סימטרית. נצטרך את המפתח בא-סימטרי ואל נצטרך
סימטרי. נצפינים עם הציבורי של המקבל.



א-סימטרי יקר לשימוש.

לכן אני קושם פותח את המפתח ואז את המייל.

PGP

צריך לזכור מאות אלגוריתמים

AES, idea א-סימטרי

RSA, elgamal סימטרי.

PGP יכול לעמוד עם אינסוף אלו.

1. אי-authentication
2. אי-confidentiality הצפנה
3. אי-compression - zip
4. email compobility
5. segmentation & reassembly

authentication

לוקחים את הודעה, עושים hash, עושים חתימה, עושים zip.

הצד השני עושה unzip, פותח חתימה, פותח את ה hash.

תמיד עושה zip כי רוצים לשלוח פחות.

confidentiality

רק פעם אחת להצפין.

לוקחת את הודעה, עושה zip, מצפין סימטרי, את המפתח מצפינים
בא-סימטרי, שולחים לצד השני

৮. নিম্নোক্ত কাজ গুলোর ফ্লো-চার্ট তৈরি করো।

ক) দুইটি সংখ্যার যোগফল বের করা।

খ) ব্যবহারকারীর কাছ থেকে ৫ টি সংখ্যা নিয়ে তাদের গড় বের করা।

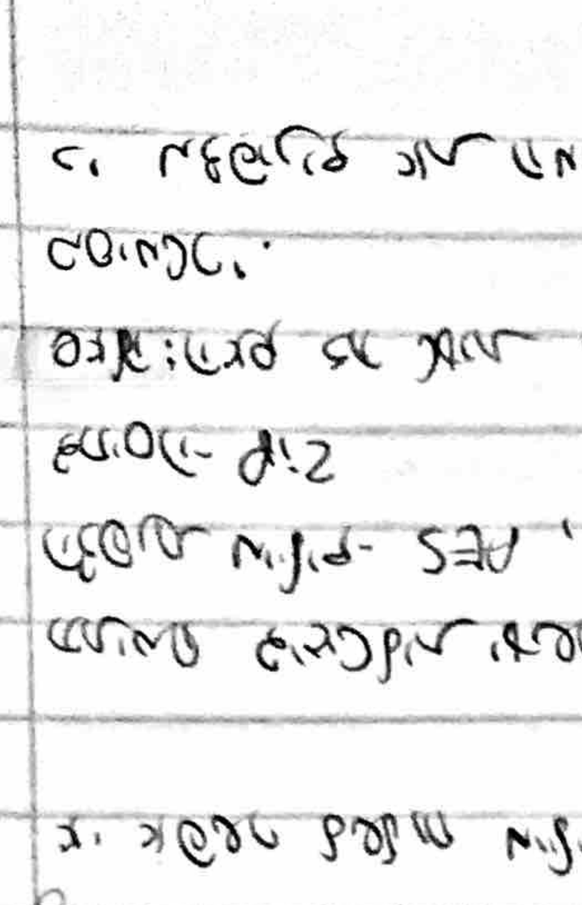
গ) ব্যবহারকারীর কাছ থেকে একটি সংখ্যা নিয়ে সেটি জোড় নাকি বিজোড় তা বের করা।

ঘ) ব্যবহারকারীর কাছ থেকে একটি সংখ্যা নিয়ে সেটি ধনাত্মক নাকি ঋণাত্মক তা বের করা।

ঙ) ব্যবহারকারীর কাছ থেকে একটি সংখ্যা নিয়ে সেটি ৩ এর গুণিতক কি না তা বের করা।

সমাধান:

ফ্লোচার্ট তৈরি করার জন্য প্রথমে ব্যবহারকারীর কাছ থেকে দুইটি সংখ্যা নিয়ে সেগুলোর যোগফল বের করতে হবে। এজন্য দুইটি ইনপুট ব্লক, একটি প্রসেস ব্লক এবং একটি আউটপুট ব্লক ব্যবহার করা হয়েছে।



প্রশ্ন ৭। কোনটি বাইনারি?

উত্তর: ৩ এবং ৪ বাইনারি সংখ্যা। কারণ এদের শুধু দুটি ডিজিট ব্যবহার করা হয়েছে। ৩ এর বাইনারি রূপ 11 এবং ৪ এর বাইনারি রূপ 100।

প্রশ্ন ৮। ফ্লোচার্ট কি?

ফ্লোচার্ট হলো একটি সমস্যা সমাধানের ধাপগুলোকে চিত্রের মাধ্যমে উপস্থাপন করার পদ্ধতি। এতে বিভিন্ন আকার ব্যবহার করে প্রোগ্রামের যুক্তি বোঝানো হয়।

প্রশ্ন ৯। ফ্লোচার্ট কেন ব্যবহার করা হয়?

ফ্লোচার্ট ব্যবহার করে প্রোগ্রামের যুক্তি সহজে বোঝা যায়। এটি প্রোগ্রামিং শেখার জন্য খুবই উপকারী।

প্রশ্নের ক- বিভাগ থেকে 2ই দিন

10.5.18 Firewalls & access control

כמו שמה שיושב בפתח
ישמש בכניסה לרשת לפני הרשת הפנימית, בין ה LAN ל WAN
ובעצם כל תעבורה אם מותר לה לכניס
שאלה: מה זה perimeter defense?
נמצא בהיקף, על כל נקודות הכניסה ויציאה של הרשת, נשים firewall
בכל נקודה כזאת. הרעיון היקפית, רק על הרשת
יש 3 סוגי firewalls:
1. personal מגן על מחשב ספציפי
2. network-מגן על הרשת
מה יתרון וחסרון שלהם יחד.
personal-מגנים בצורה שונה על משתמשים שונים שצריך לרשום
עליהם בצורה שונה, נתן אישור.
אם רוצים לרשום על הרבה מחשבים באירגון זה השה עם personal
כי זה scalability כל שיש כמות יותר גדולה כך יותר השה
לרשום עליהם.
network-מגן על כולם אבל אישה הגנה כללית.
רשת שלוב- נשים firewall על הכל, ונגן ספציפית עם personal
על מי שרוצה נכון יותר השה לפרוץ עשיני layers.
נעשה את ה firewalls שונים, מהרעיון diversity of defense (שונות הגנה).
אפשר לשים firewall בתוך הרשת על שרתים מסוימים ומי שבתוך
הרשת אם צריך לעבור דרך הfirewall שלה השרתים הספציפיים
(נשים network).
firewall מטפל על תעבורה נכנסת ויוצאת, מבחול פנימה incoming,
מבפנים לבחוץ outgoing ונשלטר.
אפשרויות:
accept-אשר את התעבורה
drop-לזרוק את התעבורה.
reject- לזרוק אבל עוד פעם משלח שהתעבורה נלקחה. כמו שיש אבל
עם הודעה שנלקחה התעבורה.

Policy actions

אפשר לשים כלל מיני חוקים. ש-2 שזה לא יתבדרו אחד עם השני.
אפשר להשביר חוקים מאיך ספציפים.
למשל, אסור לשבור בקסטו אבל כן קיד מתר.
אפשר לסדר לפי 3 שיטות
1. list black - כשיש שחורה, 2.ו.x לא יכולים להיכנס. כל השאר יכולים.
2. white list - כשיש לבן, מאושרים, ו.x יכולים להיכנס, כל השאר לא יכולים. כה לא deny חול מאמיים.
list black כה לא allow חול מלה מאמיים.

DMZ

אפשר לשים firewall, כמה מחשבים, שזה firewall ואל סוף מחשבים.
בשכבה הראשונה, שים שרתים שאנשים מבחוץ צריכים לגשת אליהם,
כו שכבת DMZ. מי שיילך לעבוד בשכבה השניה נבדקה יותר מדוקה.
אישור מבחוץ יכנס לשרת הם בשם והשרת הזה יכנס לשרת בפנים.
אם השרתים בשכבה השניה רוצים לגשת לאינטרנט הם יעשו
בעצמם.
מחשב פנימי, במקרים לגשת לרשת החיצונית, הם פונים לעצמם והוא
יביא את הכל שרצים. אפשר לחסום כבה אתרים.
מה שיש ב DMZ - שרתי mail, web, voip, DMZ.
DMZ trihomed
"כיל" אחת בחוץ, אחת בפנים ואחת ב DMZ ונשים חוקים לאן
אפשר לגשת. מה אינטרנט רק ל DMZ, מהרשת הפנימית ל DMZ
מה DMZ גם לאינטרנט ואם לרשת הפנימית
כה firewall אחד, אם מתקלקל אז הכל מתקלקל, אם פרצים לאחד אז
פרצו לכל.

types of firewall

1. stateless - כל מה שלך מגיע. נסתכל על מלה יש חוקים.
2. stateful - נסתכל על הקשר של התעבורה, תבלה שמתאים לחוקים,
לעבור, תבלה שלא מתאים לחוקים תיזרק.
תבלה שנזרקת תיא out of context, לא הגיה שם מה שמצפים

পরের প্রশ্ন প্রথমে পড়ে নাও. ব্লকের যে বালিশ কটিংগুলো রং, সেই রঙে
প্রথমে ফিট বালিশ পাড়ের কোনাগুলো বা ব্লকের
প্রশ্নটা হচ্ছে মোট সাইজটা ২-নং ব্লকের পাশাপাশি রং,
পাড়ের কিছু অংশ ও ব্লকের প্রয়োজনীয় দৈর্ঘ্য, প্রস্থ, ডায়াগনাল ইত্যাদি
ফিটিংস কটিংগুলো ও ডায়াগনাল বালিশ কাটার পরে পাড়ের
ফ্যাব্রিক কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে
ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

ফ্যাব্রিক ব্লক, ব্লকের সাইজ

পাড়ের প্রস্থ প্রথমে পুরো এক ব্লকের যে বালিশ কটিংগুলো রং, সেই রঙে
প্রথমে ফিট বালিশ পাড়ের কোনাগুলো বা ব্লকের
প্রশ্নটা হচ্ছে মোট সাইজটা ২-নং ব্লকের পাশাপাশি রং,
পাড়ের কিছু অংশ ও ব্লকের প্রয়োজনীয় দৈর্ঘ্য, প্রস্থ, ডায়াগনাল ইত্যাদি
পাড়ের ফ্যাব্রিকগুলো কটিংস ও ডায়াগনাল বালিশ কাটার পাড়ের
ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

ফ্যাব্রিক ব্লক, ব্লকের

পাড়ের প্রস্থ প্রথমে পুরো এক ব্লকের যে বালিশ কটিংগুলো রং, সেই রঙে
প্রথমে ফিট বালিশ পাড়ের কোনাগুলো বা ব্লকের
প্রশ্নটা হচ্ছে মোট সাইজটা ২-নং ব্লকের পাশাপাশি রং,
পাড়ের কিছু অংশ ও ব্লকের প্রয়োজনীয় দৈর্ঘ্য, প্রস্থ, ডায়াগনাল ইত্যাদি
পাড়ের ফ্যাব্রিকগুলো কটিংস ও ডায়াগনাল বালিশ কাটার পাড়ের
ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

ফ্যাব্রিক ব্লক, ব্লকের

* পাড়ের রং

বালিশ - কটিংস

ডায়াগোনাল - বালিশ

১. ব্লকের সাইজ ২-নং ব্লকের পাশাপাশি রং, পাড়ের কিছু অংশ ও ব্লকের প্রয়োজনীয় দৈর্ঘ্য, প্রস্থ, ডায়াগনাল ইত্যাদি পাড়ের ফ্যাব্রিকগুলো কটিংস ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

২. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৩. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৪. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৫. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৬. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৭. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৮. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

৯. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

১০. ব্লকের সাইজ ও ডায়াগনাল বালিশ কাটার পাড়ের ফিনিশিং পাড়ি কী, হিসেব করে ফিটিংস ও পাড়ি, ডায়াগোনাল কি হবে

של SQL injection, אין authentication, אפשר לעשות address spoofing.

stateful מסתכל איך הגיעה החבילה, עוקב אחרי ה sessions. נסתכל שכל חבילה שייכת ל session שאושר, נסתכל שהחבילה תקינה ולא out of context.

שים לב לפורטים, לסוג התקשורת (TCP | UDP).

נבדק כמה פרמטרים, אם חבילה שלפחות אחד שונה כזה out of context ונזרקים אותו.

מסתכל על IP מקור ויעד ופורט מקור ויעד.

circuit level gateway הוא TCP proxy. רק לTCP ולא אפליקציה.

application level gateway הוא בעצם proxy. אם יש הרבה אפליקציות שנכנסות ויוצאות מהארגון, לכל אחת יש proxy.

אפשר להריץ מחשב אחד עם תהליכים לכל אחד מהפרוטוקולים.

אם הלקוח רוצה לפנות החוצה, הוא פונה ל proxy שפונה החוצה, מקבל תשובה ומעביר ללקוח. וגם הפוך (מבחוץ פנימה)

proxy נפרד זה לא בהכרח מחשב נפרד.

לכל שרת proxy משלו. יש http proxy, proxy

אפשר לרשום את כל ה traffic, ככל שעושים יותר עבודה יותר קשה.

bastion host

bastion - מבצר.

הגדרה- תהליכים ספציפיים במחשב ספציפי. רק אז אפשר לתקוף.

host מוקשח שמכינים להגן עליו. משאירים תהליכים ספציפיים בהתקנת מערכת ההפעלה.

נשתמש ב bastion host בתוך firewall.

Single homed - יש לו רק רשת אחת. שם אותו על הרשת, ושמים מחשב מוקשח, בכניסה שמים firewall פשוט (stateless), כל דבר שהולך אל bastion host נכנס, שום דבר לא יוצא החוצה, רק bastion host והוא גם משמש כ proxy. פשוטו firewall כפול שצריך לעבור את שניהם. אחד firewall ואחד מחשב מוקשח.

ה bastion host יחליט מה נכנס ומה יוצא. יש גם packet וגם application

dual home - בעל 2 כרטיסי רשת. מפריד בין הבחוץ לבפנים.

কোনো প্রোগ্রাম চলমান অবস্থায় থেকে প্রসেস।
কাজ শুরু হলো।
কাজ শেষ হলে মেমরিটি ফ্রি হবে না প্রসেস কাজ শেষ করে মেমরি
ফ্রি করে দিয়ে যেতে পারে মেমরি'তে মেমরি।
সাত ফ্রি যাবে?

Windows
অপারেটিং সিস্টেম বলতে কি বুঝায়, উইন্ডোজ
কম্পিউটার ব্যবহার করার জন্য এমন একটি সফটওয়্যার যা কম্পিউটার
এবং ব্যবহারকারীর মধ্যে যোগাযোগ রক্ষা করে থাকে।
Windows, output, input প্রভৃতি
ডিভাইস ড্রাইভার, প্রিন্টার ড্রাইভার, কিবোর্ড ড্রাইভার, মাউস ড্রাইভার
ইত্যাদি ড্রাইভার। এগুলো সব ডিভাইস ড্রাইভার কিন্তু প্রিন্টার ও
কিবোর্ড ড্রাইভার আলাদা করে ডিভাইস ড্রাইভার হিসেবে কাজ করে।

(MIT) মেসাচুসেটস ইনস্টিটিউট অফ টেকনোলজি
এমআইটি'র কম্পিউটার সায়েন্স এন্ড আর্টিফিশিয়াল ইন্টেলিজেন্স
ল্যাবরেটরি (CSAIL) এর গবেষকরা একটি নতুন পদ্ধতি উদ্ভাবন করেছেন
যা বড় ভাষা মডেলগুলিকে (LLMs) আরও নির্ভুল এবং প্রাসঙ্গিক
উত্তর তৈরি করতে সাহায্য করবে।
এটি কাজের জটিলতা অনুযায়ী স্বয়ংক্রিয়ভাবে একাধিক মডেল
নির্বাচন করে এবং প্রশ্নের উত্তর যাচাই করে দেখে যে উত্তরটি
সঠিক কিনা।
এর ফলে ভুল তথ্য কমে আসবে এবং জটিল কাজে
নির্ভুলতা বাড়বে।

Windows
উইন্ডোজ হলো মাইক্রোসফট কর্তৃক তৈরি একটি অপারেটিং
সিস্টেম যা কম্পিউটার ও মোবাইল ডিভাইস পরিচালনা করে থাকে।
উইন্ডোজ ব্যবহার করে ব্যবহারকারী ফাইল তৈরি, ডকুমেন্ট
সম্পাদনা, ইন্টারনেট ব্রাউজিং, গেম খেলা এবং বিভিন্ন সফটওয়্যার
চালানো সম্ভব হয়।
উইন্ডোজ ব্যবহারকারীর জন্য সহজ ইন্টারফেস তৈরি করে।
উইন্ডোজ ব্যবহারকারীর জন্য বিভিন্ন ফিচার যেমন ফাইল ম্যানেজমেন্ট,
ইন্টারনেট ব্রাউজিং, মাল্টিমিডিয়া প্লে, অফিস অ্যাপ্লিকেশন ও
গেমিং সাপোর্ট দেয়।
উইন্ডোজ ব্যবহারকারীর নিরাপত্তা বজায় রাখে এবং ফাইল
ম্যানেজমেন্ট করে থাকে।

שאלות: שת 67. 68-70 תשובות

יש 3 סוגי firewall. על כל התבשה נשאל האם ה firewall הנ"ל פשוט
כדי לדעת, אם כן איך, אם לא מי ואיך.

1. מתבצע flood מעל מרשת חיצונית
flood מעל מכונה למשת לסם.
אי אפשר עם stateless.
ב stateful כדי מובר פרק ה firewall. רואים שיש SYN, SYN ACK
ואין ACK. הוא מספיקה.

2. smurf attack
smurf - שולחים "ברדסים", ping בבroadcast עם מקור של מישהו
ובולם מגיע לו.

מישהו עשה spoofing ושם
במקום מישהו אחר. תוקפים מישהו מבפנים, המקום של מישהו מבפנים.
stateless יכול, הוא לא יאפשר ping שבעצם הוא broadcast.

3. לעדכן אפליקציה קשה לדוגמא BitTorrent
בגלל שרוצים לעדכן אפליקציה מסוימת צריך application firewall
לכן כדי השעה, צריך לראות מה האפליקציה שרצה.

שקף 71

	action	our IP	our port	their IP	their port	comment
1	allow	*	*	*	80	
	allow	*	*	*	https port (443)	
2	block	162.1.1.1	80	113.17.19.5	*	
	block	162.1.1.1	80	148.1.23	*	
	allow	162.1.1.1	80	*	*	
3	allow	162.1.1.2	SSH, 22	*	*	
4	block	*	*	*	*	

4. אם עמדנו הכל מהתחלה (שים לפני 3) לא יופעל לדברים אלינו
כי המענה הכל מאפל.

প্রশ্ন ১৪.

সংখ্যাটি কিভাবে লিখলে বড় মান প্রথম এবং ছোট মান শেষে থাকবে। * এর পর প্রথম বড় মান নিয়ে পরবর্তী বড় মান নিয়ে এবং সবশেষে ছোট মান নিয়ে

Access control

יתן למשתמש מה הוא יכול לעשות

אובייקט -מה שצריך לגשת אליו. קבצים, שרתים

subject -מי ניגש. משתמש, אפליקציה.

2 שיטות:

1. matrix
2. RBAC

access matrix

בשורות האובייקטים, בעמודות
מי יכול לעשות, בפנים איזה
הרשאות יש לו.

subject ....
object
⋮

מחשב סורק לפי שורה שורה או עמודה עמודה.

1. capability list סורק לפי שורה, לפי subject.
2. access control list (ACL) - סורק לפי עמודות, לפי אובייקטים. בקובץ X
יש למשתמשים A,B,C הרשאות 1,2,3. סורקים בד"כ עם זה
ו סורק לפי כל משתמש רואים איזה הרשאה יש לו.

RBAC

role based access control.

נותנים הרשאות בהתאם לתפקיד.

ככל שעולים בהיררכיה יש יותר הרשאות. מקבלים הרשאות מסוימות
לפעולות הרגילות

השיטה חוסכת את זה שכשעוברים תפקידים ומבקשים עוד
הרשאות לתפקיד החדש ולא מורידים את הקודם. בRBAC משנים לפי
תפקיד, קודם מורידים הרשאות ואז מעבירים לתפקיד והרשאות חדשות.

יש XML ל access control, נקרא XMCAL.

24.5.18 intrusion detection - IDS

יש 2 מגנים:
1. firewall
2. IDS

intrusion - קשה להגדיר. כל דבר רע. כל דבר שמישהו עושה ולא מורשה לעשות.

התקפות
mapping -
סריקת פורטים, NMAP, sniffing
IP spoofing -
שליחת חבילה בשם מישהו אחר

DoS, DDoS
ב DDoS, התוקף משתמש בבוטים לתקוף.

sniffing - האזנה. נקראו גם monitoring. נעשה ע"י כלים שנמצאים באינטרנט.

הגנה
עד עכשיו הכרנו firewall.
יש גם אבטחה ברמת מערכת ההפעלה - שם משתמש וסיסמא. אפשר גם רשימת access שנותן הרשאות למשתמש.
יכולות להיות חבילות תקינות אבל שילוב שלהם זה בעצם התקפה ואת זה firewall לא יזהה.

IDS
יכול לזהות התקפה.
יש לו DB של התקפות, אם יש פעולות שנראות כמו התקפות בשבילו הוא מתריע בהתקפה.
לIDS יש sniffers כדי לראות את כל החבילות ברשת כדי לזהות אם יש התקפה או לא.
שמים IDS אחרי הfirewall, מתוך הרשת. כך התקיפה שלו תהיה

* প্রশ্ন ১৫ - ডিফিং মেশিন

Types of Intruders

১। External Penetrator - যারা বাইরে থেকে নেটওয়ার্ক ব্যবহার করে ঢুকতে চায়।

২। Internal Penetrator - যারা নেটওয়ার্কের ভেতরে অনুমতি আছে কিন্তু অনুমতির বাইরে কাজ করতে চায়।

৩। Clandestine User - যারা নেটওয়ার্ক প্রশাসক বা সিস্টেম কন্ট্রোল নিয়ে নেটওয়ার্ক ব্যবহার করে।

Intruder Behavior

১। Masquerader - অন্য কারো পরিচয় ব্যবহার করে ঢুকে পড়া, যেমন হ্যাকার।

২। Misfeasor - অনুমতি আছে কিন্তু অপব্যবহার করে, যেমন অসৎ কর্মচারী।

৩। Clandestine User - নিয়ন্ত্রণ নিয়ে লগ মুছে ফেলা, যেমন সাইবার অপরাধী।

Intrusion Detection - IDS

নেটওয়ার্ক বা সিস্টেমে অনুমতিহীন প্রবেশ শনাক্ত করার সিস্টেম। IDS নেটওয়ার্ক ট্রাফিক বা সিস্টেম লগ বিশ্লেষণ করে সন্দেহজনক কার্যকলাপ খুঁজে বের করে।

IDS Types

১। Network based IDS - নেটওয়ার্ক ট্রাফিক বিশ্লেষণ করে।

২। Host based IDS - একটি নির্দিষ্ট সিস্টেমে চলে এবং সিস্টেম লগ বিশ্লেষণ করে।

Signature Based IDS

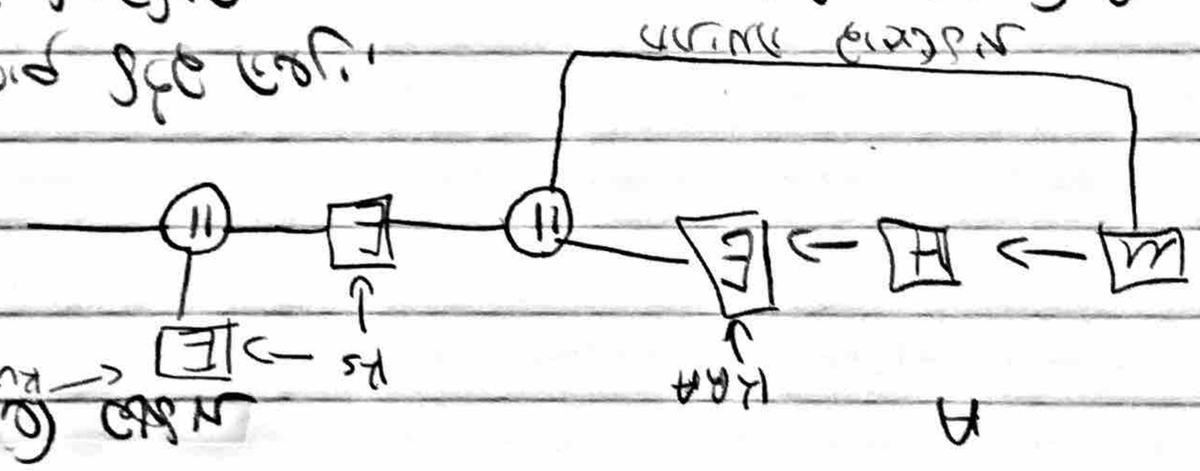
পরিচিত আক্রমণের প্যাটার্ন খুঁজে বের করে। দ্রুত কাজ করে কিন্তু নতুন আক্রমণ ধরতে পারে না।

Anomaly Based IDS

স্বাভাবিক আচরণের তুলনায় অস্বাভাবিক কার্যকলাপ শনাক্ত করে। নতুন আক্রমণ ধরতে পারে কিন্তু ভুল সতর্কতা বেশি দেয়।

সার্কিট ডায়াগ্রাম :

ধরা যাক, একটি সংখ্যা ডাটা রেজিস্টারে আছে। সংখ্যাটি যোগ করতে হবে। যোগ করার জন্য একটি ALU ব্যবহার করা হয়। ALU-র দুটি ইনপুট থাকে। একটি ইনপুট সরাসরি ডাটা রেজিস্টার থেকে আসে। অন্য ইনপুট আসে টেম্পরারি রেজিস্টার থেকে। যোগফল আবার ডাটা রেজিস্টারে ফিরে যায়। টেম্পরারি রেজিস্টারের ডাটা সবসময় ALU-র ইনপুটে থাকে। ALU-র আউটপুট সরাসরি ডাটা রেজিস্টারে যায়। ALU-র আউটপুট সরাসরি ডাটা রেজিস্টারে যায়।



LDA - A রেজিস্টারে সংখ্যা লোড করা
LS - ডাটা রেজিস্টারে সংখ্যা লোড করা
A- প্রসেসরে রেজিস্টারে সংখ্যা লোড
W - প্রসেসর