# Hacker 101

CODE BY ALEKS/ARIS/HAYLEY/JACOB/YURI

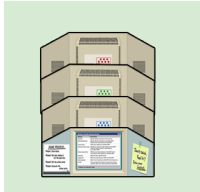**4 PLAYERS**
20–30 MINS
4 ROUNDS

Welcome! Hacker 101 is an action programming & dice rolling competitive game. In this game, four players are hackers trying to attack three servers together to obtain as much confidential information as you can from the governments' intelligence agency. Remember, you are not friends, make action to benefit yourself only!
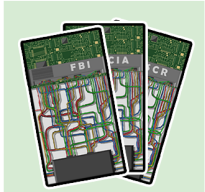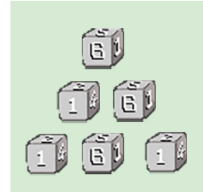
3x SRV	4x COMPUTERS	22x ACTION CARDS	4x EVENT CARDS	6x LOAD DICE	20x DATA CUBES	12x MARKERS

GAME GOAL

// The entire game will be played for 4 rounds, with each player in turn becoming the starting player for each round according to the clockwise direction.

// If the sum of data number is less than or equal to the number of dice in the server, the player receives one point for each dice the player has in the SRV.

// If the total number of DATA is greater than the number of dice in the server, then no player scores a point

// The player who gets the most points wins.

SET UP

// The youngest player becomes the first player ,play clockwise.

// Draw an event card and according to the card place the dice into different servers

// Each player pick a color, take 3 same color markers 、5 data tokens and 5 action cards.

HOW TO PLAY

Phase1: Draw a event car and put data tokens into SRV.

Draw an event card and according to the card place the dice into different servers.The youngest player is the first player. Starting with the first player, players act clockwise. How to place the data : all players choose the amount they want to put in their hands and put it into the server together. Put the data in each server in turn according to the order on the event card

Note: Players can choose not to put any data in a server, but at the end they must put all the data into the servers.

Phase2: Reroll all the load dice.

The first player reroll all the dice in each server

Phase3: Put the action cards.

Players can place an action card on the server.Player can place it on any server, but if any action card has already been placed on a server, player can follow it and place it next. When placing it, player need to place a maker on the card. It will take three place rounds, which means player will place a total of three action cards on servers

Phase4: Execute the action cards.

Players need to follow the server order to reveal the cards.The first placed card being revealed first in a server. When executing a card, the owner of the card has to make a decision based on the card effect.

Note: Player can use the objection card from your hand to cancel any card effect when it is revealing

Note: Reveal Action Cards according to the order number on the slot. Start with the first SRV on the Event Card. After executing cards, players count their points to see whether the SRV survived or crashed down. A SRV is survived if and only if the Num of data is less than SRV Firewall Points, or it is down.

DESIGN STATEMENT

The game was adapted from the bath game, which had many problems, the first of which was the lack of decision–making space for the player. The second point is that the game's card system is designed too rigidly, so that players' choices will be the same. So we modified the card system by turning the player's turn to draw a hand into a public area that would show three cards at a time, and each turn the player would need to choose one of those cards to add to their hand. This worked pretty well, as it was equivalent to other players being able to know partially what the other player's hand was, and everyone would keep an eye on the cards they wanted for fear of being snatched up by someone else first, increasing competition between players.

However, after doing so, it was found that the players' choices were still too narrowed, and even when choosing cards from the public area, players would all choose the one with the strongest ability. So we wondered what would happen if we just kept the core dice throwing mechanic and combined it with other mechanics.

After trying different combinations, we chose the Action Programming mechanic, which has a high degree of randomness at its core, and after playing the masterpiece game of the genre, Colt Express, we decided that this was the kind of relaxed atmosphere we wanted.

The game has many playtests, each time we tried to make some changes while maintaining the core of the game.

So we went ahead and designed different bath tubs and cards, and added event cards to increase repeatability, which allowed the game to change from game to game. But this time it turned out that the game's narrative needed to be redesigned, and we tried a lot of themes, like survival ships, subways, etc., but finally settled on haker because every game mechanic could be well explained in that theme and it sounds cool.

We put a lot of effort into numerical modifications, for example, at the beginning players generally thought there was a big problem that the game was too complicated to calculate. Because at that time each player had 9 data, and there were 4 servers on the field, players thought that there was a little too much to calculate. So after repeated testing, we finally settled on 5 data and reduced the number of servers to 3.

At first, we had three action card caps for each server, but some players said that this would have too much of an impact on the game's parity and make players feel like they couldn't use the action they wanted to use on the correct server. So we set the server action cards to have no cap, and this worked well after testing.

We didn't want to make a game that required serious thinking from the beginning, so the whole creative process was about how to use randomness to generate fun, and we did a lot of experimenting, and although it wasn't perfect, we played and laughed there every time we tested it, and in a way, that was our biggest success.