**CVE and small business disclosure notes:**

I. Background

In 2020 I was the first person to bring a pinball machine to a security conference for the purpose identifying a vulnerability. The game I brought was a Jersey Jack Pinball game with the title Dialed in. I brought this game because it was the most technologically advanced game ever made and introduced a mobile application with Bluetooth that could interact with the game. I brought this game to a security conference to introduce the BSides security community to pinball for fun and to identify any vulnerabilities that could impact a competitive game. As a hobbyist operator, owner of the DialedIn game and competitive world ranked player I wanted to determine if the game had any vulnerabilities that could be exploited. This paper was prepared to document the relatively simple steps from beginning to end that were used to determine that an exploitable vulnerability/bug was present in the game.

**II. Discovery**

The Bluetooth issue we discovered during the conference did not require detailed planning, special tools, hacking of software or introduction of third-party tools that would normally be used during a formal penetration test. This was effectively low-hanging fruit and I believed at the end of the event that any player or pinball threat actor within **proximity of one of these games** and a **downloaded version of the mobile app** could exploit this vulnerability.

The scenario we observed occurred only during multiplayer games and code base 1.73. The result allowed another player or individual **with a previously paired** device to impact, or interrupt an active player during an active multiplayer game. The threat actor would need to be within range of approximately 33 feet without amplification, which could extend the range of these attacks significantly. The condition seemed to be limited to previously paired devices which indicated that the pinball machine may store a history of paired devices. We ran several scenarios on different devices that included Android and IoS phones to recreate this condition and we felt that other Bluetooth vulnerabilities were likely present that we were not able to consistently repeat. This was not an unexpected result given the innovative nature of the mobile application. As someone who has played competitive pinball at the highest levels, I believe the impact on to $100k+ e-sports Pinball tournament could be significant and have recommended changes to mitigate or eliminate this risk to competitive play.

*Note: This vulnerability was tested and validated on version **1.73** of the code and I do not see any notes in version **1.75** that indicate that any security updates would have changed this but I will need to test again against version **1.75**. I also want to note that the mobile app is no-longer supported or available in the app stores, which may be an indirect result of my research?*
*REFERENCE CODE Changelog:*
https://marketing.jerseyjackpinball.com/di/di_changelog.txt

**III. Reporting**

After discovering this issue, I emailed an individual at the pinball company and did not receive a response. A few months later I noticed that the apps were removed from the App store. A release on October 18 2021 deprecated support for mobile application. I assumed that my disclosure or other similar disclosures may have resulted in this app removal. I will note that:

A. Older versions of the app can still be potentially found, downloaded or side-loaded outside of the app store.
B. If the Bluetooth dongle remains in place, the game may still be vulnerable in other ways (e.g. add list of tools and exploits against BT here)
C. Individuals with a deprecated version of the app may have migrated it to their new phones(I confirmed that this is possible with android, but I was unable to get the deprecated app to pair to the game via the app on initial testing)

The challenges I experienced in reporting this vulnerability were likely the same issues that others have experienced when reporting to a small business. Small businesses don't have CISO's, security teams or a bug bounty program. Most small businesses also don't have a Security@yourcompany.com email. I have published some guidance that I am circulating/attempting to the large pinball companies that hopefully will help to position them going forward. As pinball enters the world of IoT, secure by design and the ability to harness the power of researchers will be an important factor in securing the game.

## IV. Assigning a CVE

The process to assign a CVE is relatively easy https://cveform.mitre.org/ and I thought logging the first CVE for a pinball machine might be worthy of a submission. I may or may not go through the formality in advance of the talk.

Exploit-DB submission: In Process
CVE submission: on hold until testing of version 1.75

## V. Lessons Learned

a. Contacting Vendors is very difficult. Reaching out to executives in companies won't work unless you prove a positive business case. The next time I find a vulnerability in a pinball machine I will likely directly connect to a developer and establish a partnership that will lead to safer pinball products for everyone.
b. Submitting a CVE is easy but I wish I had formally documented every detail including versions of IOS and Android and app versions and found a way to record the BT traffic during the initial testing.

## VI. Summary and next steps

Working to build relationships with Devs from all 3 pinball companies now. "Skynet development continues…"

## APPENDIX

## VII.

Pinball businesses are small businesses and small businesses don't have vulnerability disclosure programs. During the testing it was determined that previously paired devices could interrupt or interfere with currently paired devices.

1. Remove the Bluetooth Dongle from pinball machines during tournament play!
2. Release updated in code that allows an operator to Switch off the device's Bluetooth to prevent other devices from pairing with it.
3. Release updated code that prevents Bluetooth pairing during a multi-player game.
4. Limit the number of apps that have access to the device's Bluetooth connection to 1 or less.
5. Release code update to prevent in-game pairing.
6. Release code update to show on screen paired devices to the active player.
7. Clear a history/memory of previously paired devices after a game completes.
8. Keep the device updated with latest code, security updates or patches. Turn on auto update.
9. Establish a dedicated email for reporting bugs/vulnerabilities security@pinballcompany.com

---

Actual CVE submission goes here if/when I submit