
# WebNFC Bridge

WebNFC[1] Use-case for Secure Interaction Between
an Untrusted Web-page and a Connecting Mobile Device

The following *conceptual specification* builds on the same core as the Web2Native Bridge[2]. In fact, the *intention* is that invoked native applications would be *identical* for both schemes.

On the Web-side there would be minor differences, since NFC and local application invocation have distinct JavaScript interfaces, whereas the actual *application code* should be identical, including error handling.

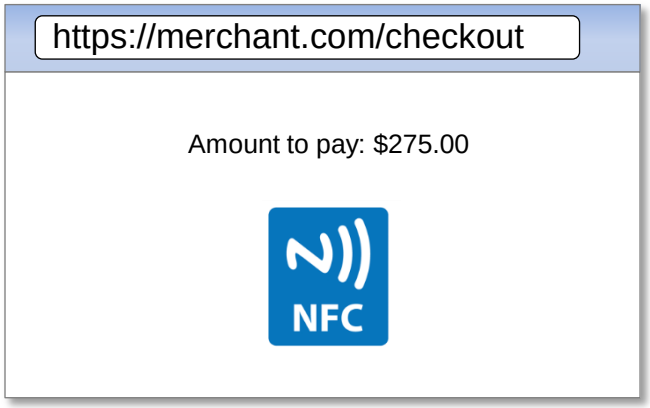
To create a good user experience, the actual transactions are supposed to be carried out over BLE (Bluetooth Low Energy). That is, NFC is only used for BLE paring and naming the application to invoke.

Note that this specification does not include a security element since such functionality can be supplied in many different ways when needed.

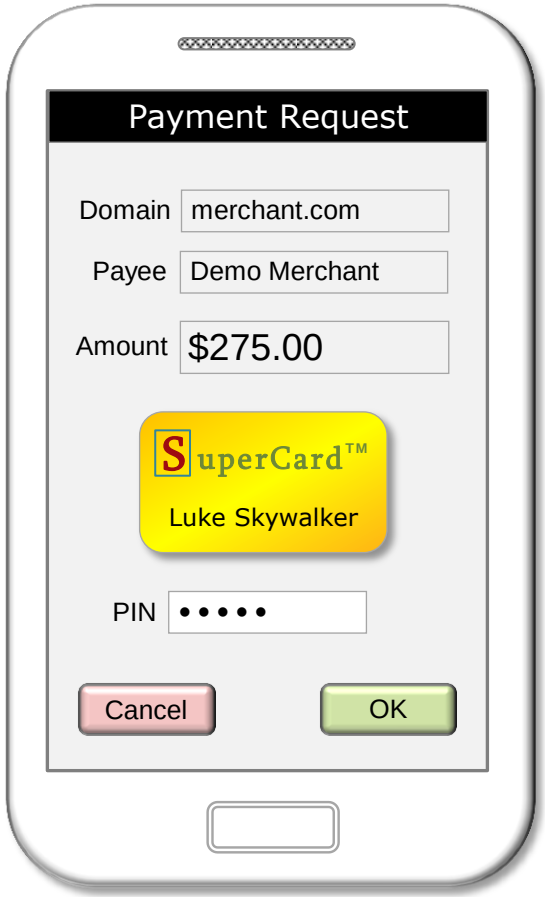
1. https://www.w3.org/community/web-nfc/
2. https://cyberphone.github.io/openkeystore/resources/docs/web2native-bridge.pdf

# WebNFC Bridge – Typical Use Case

① User interacts with a Web application on a PC, POS terminal, Vending machine, etc.

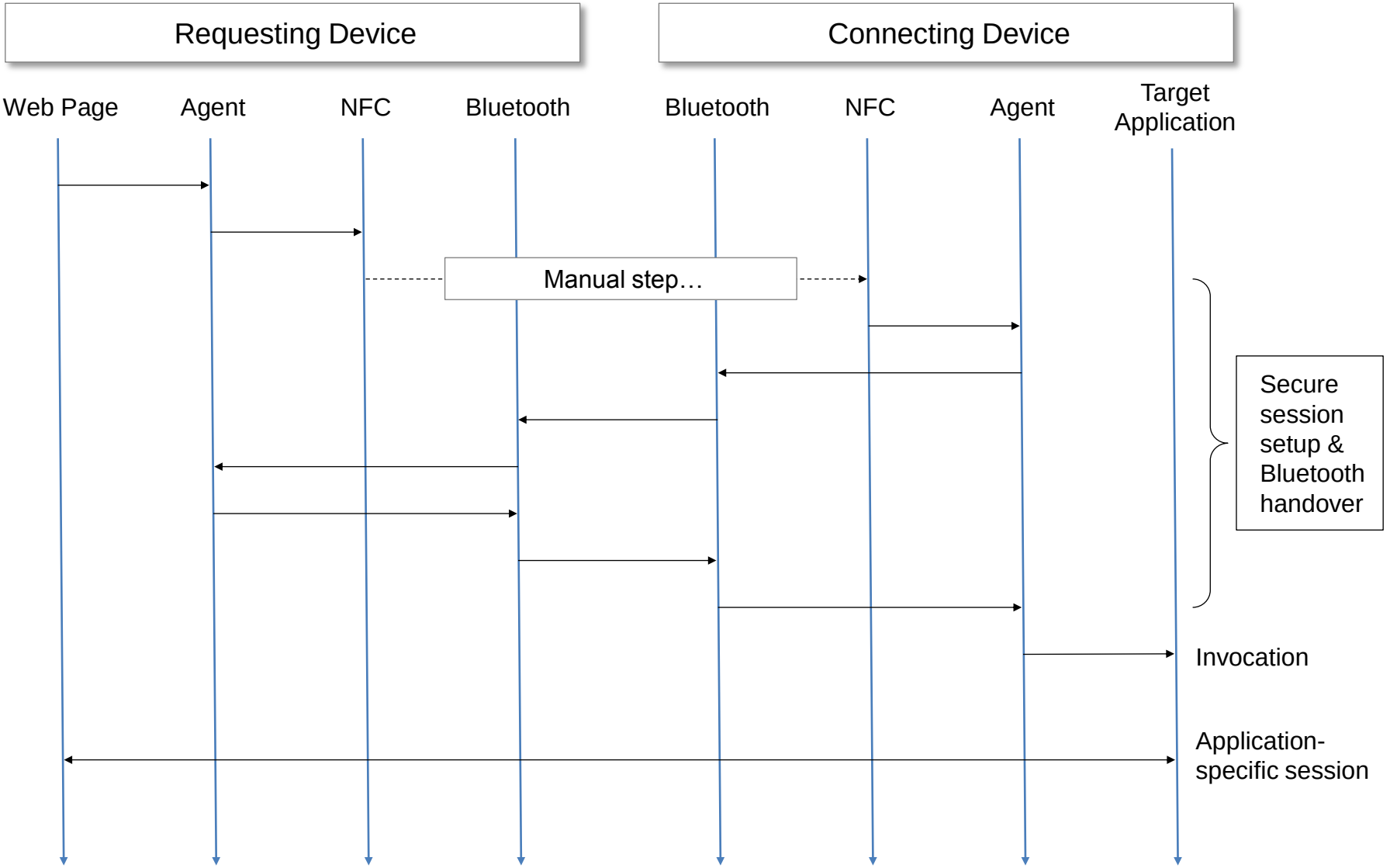



② User performs the NFC connection




③ User finishes the request in the securely connected mobile device

# WebNFC Bridge – Sequence Diagram



# WebNFC Bridge – Applications

- Secure Web Payments
- User Authentication
- Supplying User-data
- Games
- Virtual Passports & Visas
- Boarding Cards
- Interaction with TVs including remote control
- You name it…