```
  QSPI_Read_Product_Headers((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);
}
while ( BYTE1(configuration_script) != 1 );
WDOG_feed_ff();
if ( check_current_fw_addr_and_update_addr(product_img_offsets) )// differing offsets mean we have an update, enter
{
  if ( !SECUREBOOT_check_and_validate(
          product_img_offsets[1],
          (FW_ImageHeader *)&dword_2003C950,
          (int)product_img_offsets,
          (struct_configuration_script_ptr *)&configuration_script) )// / returns 1 if secure boot is not enabled
  {
    QSPI_process_product_update((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);// re
    WDOG_Pet_1c34();
  }
}                                               // no update path
                                                // this returns 0 from [8] offset check
                                                //   if ( !(*qspi_data_value_ptr)[8] )
                                                //     return 0;
else if ( !SECUREBOOT_check_and_validate(
             product_img_offsets[0],
             (FW_ImageHeader *)&dword_2003C950,
             (int)product_img_offsets,
             (struct_configuration_script_ptr *)&configuration_script) )// returns 1 if secure boot is not enabled
{
  WDOG_Pet_1c34();                              // we hit this
}
SECUREBOOT_CHECK_key_revocation(dword_2003C950);
sub_A6E((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);
WDOG_feed_ff();
if ( !SECUREBOOT_CHECK_setup_QSPIC_CTR_195C(
        (struct_configuration_script_ptr *)&configuration_script,
        product_img_offsets) )
  WDOG_Pet_1c34();
OTPC_standby();
Cache_setup_qspi_cache(dword_2003C950, product_img_offsets);
write_to_sysram(dword_2003C950, product_img_offsets);
return RESET_to_REMAP_ADR_val(2u);              // SW Reset to QSPI Flash
}
```


CVE-2024-25076

CVE-2024-25076

```
    QSPI_Read_Product_Headers((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);
  }
  while ( BYTE1(configuration_script) != 1 );
  WDOG_feed_ff();
  if ( check_current_fw_addr_and_update_addr(product_img_offse
  {
    if ( !SECUREBOOT_check_and_validate(
            product_img_offsets[1],
            (FW_ImageHeader *)&dword_2003C950,
            (int)product_img_offsets,
            (struct_configuration_script_ptr *)&configuration_script) )// / returns 1 if secure boot is not enabled
    {
      QSPI_process_product_update((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);// re
      WDOG_Pet_1c34();
    }
  }                                             // no update path
                                                // this returns 0 from [8] offset check
                                                //   if ( !(*qspi_data_value_ptr)[8] )
                                                //     return 0;
  else if ( !SECUREBOOT_check_and_validate(
              product_img_offsets[0],
              (FW_ImageHeader *)&dword_2003C950,
              (int)product_img_offsets,
              (struct_configuration_script_ptr *)&configuration_script) )// returns 1 if secure boot is not enabled
  {
    WDOG_Pet_1c34();                            // we hit this
  }
  SECUREBOOT_CHECK_key_revocation(dword_2003C950);
  sub_A6E((struct_configuration_script_ptr *)&configuration_script, product_img_offsets);
  WDOG_feed_ff();
  if ( !SECUREBOOT_CHECK_setup_QSPIC_CTR_195C(
          (struct_configuration_script_ptr *)&configuration_script,
          product_img_offsets) )
    WDOG_Pet_1c34();
  OTPC_standby();
  Cache_setup_qspi_cache(dword_2003C950, product_img_offsets);
  write_to_sysram(dword_2003C950, product_img_offsets);
  return RESET_to_REMAP_ADR_val(2u);            // SW Reset to QSPI Flash
}
```

ENCRYPTION

Key Indexes
- **Index into OTP for encryption engine**

Nonce
- **User defined Nonce**
- **Avoid IV reuse**

AES-CTR mode enables fast/arbitrary block decryption
AES-CTR provides no auth (malleable)