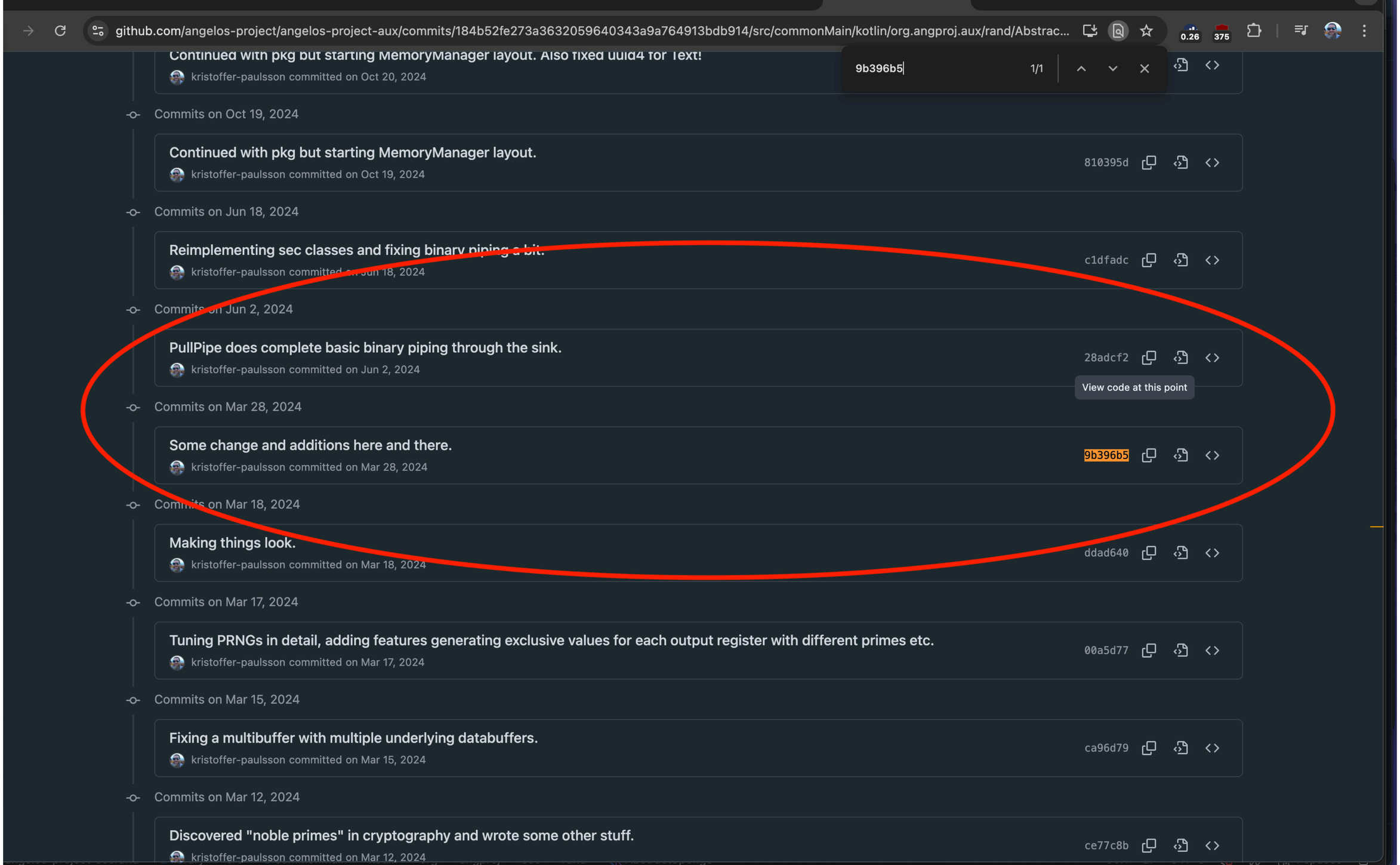
# Angelos Project attacked

## Someone tried to inject code in one of the critical cryptographic component of Angelos Project. Discovered current date for earlier happening.

angelos-project-aux/library/src/commonMain/kotlin/org.angproj.aux/rand/AbstractSponge.kt

https://github.com/angelos-project/angelos-project-aux/blob/master/library/src/commonMain/kotlin/org.angproj.aux/rand/AbstractSponge.kt

Kristoffer Paulsson, 2025-09-14


github.com/angelos-project/angelos-project-aux/commits/184b52fe273a3632059640343a9a764913bdb914/src/commonMain/kotlin/org.angproj.aux/rand/Abstrac...
9b396b5 1/1
Continued with pkg but starting MemoryManager layout. Also fixed uuid4 for Text!
kristoffer-paulsson committed on Oct 20, 2024
Commits on Oct 19, 2024
Continued with pkg but starting MemoryManager layout.
kristoffer-paulsson committed on Oct 19, 2024
810395d
Commits on Jun 18, 2024
Reimplementing sec classes and fixing binary piping a bit.
kristoffer-paulsson committed on Jun 18, 2024
c1dfadc
Commits on Jun 2, 2024
PullPipe does complete basic binary piping through the sink.
kristoffer-paulsson committed on Jun 2, 2024
28adcf2
View code at this point
Commits on Mar 28, 2024
Some change and additions here and there.
kristoffer-paulsson committed on Mar 28, 2024
9b396b5
Commits on Mar 18, 2024
Making things look.
kristoffer-paulsson committed on Mar 18, 2024
ddad640
Commits on Mar 17, 2024
Tuning PRNGs in detail, adding features generating exclusive values for each output register with different primes etc.
kristoffer-paulsson committed on Mar 17, 2024
00a5d77
Commits on Mar 15, 2024
Fixing a multibuffer with multiple underlying databuffers.
kristoffer-paulsson committed on Mar 15, 2024
ca96d79
Commits on Mar 12, 2024
Discovered "noble primes" in cryptography and wrote some other stuff.
kristoffer-paulsson committed on Mar 12, 2024
ce77c8b

Files

9b396b5

Go to file t

.idea
gradle
src
commonMain/kotlin/org.angproj....
codec
fsm
io
num
pkg
rand
AbstractSmallRandom.kt
AbstractSponge.kt
AbstractSponge1024.kt
AbstractSponge256.kt
AbstractSponge512.kt
InitializationVector.kt
reg
sec
ui
utf
util
commonTest
jsMain
jvmMain



kristoffer-paulsson Some change and additions here and there.

9b396b5 · last year History

Code Blame 58 lines (49 loc) · 1.72 KB

Raw

```kotlin
/**
 * Copyright (c) 2024 by Kristoffer Paulsson <kristoffer.paulsson@talenten.se>.
 *
 * This software is available under the terms of the MIT license. Parts are licensed
 * under different terms if stated. The legal terms are attached to the LICENSE file
 * and are made available on:
 *
 *      https://opensource.org/licenses/MIT
 *
 * SPDX-License-Identifier: MIT
 *
 * Contributors:
 *      Kristoffer Paulsson - initial implementation
 */
package org.angproj.aux.rand

import org.angproj.aux.util.DataBuffer
import org.angproj.aux.util.EndianAware
import org.angproj.aux.util.floorMod

/**
 * AbstractSponge is a class that circumvents unnecessary
 * bugs when implementing a secure random sponge construction.
 * */
public abstract class AbstractSponge(spongeSize: Int = 0, public val visibleSize: Int = 0) : EndianAware {

    protected var counter: Long = 0
    protected var mask: Long = 0
    protected val sponge: LongArray = LongArray(spongeSize) { InitializationVector.entries[it].iv }
    public val byteSize: Int = visibleSize * Long.SIZE_BYTES

    init {
        require(visibleSize <= spongeSize) {
            "Visible size must be equal or less than the number of sponge variables."
        }
    }

```

Files

28adcf2

Go to file t

.idea
gradle
src
commonMain/kotlin/org.angproj....
buf
codec
fsm
io
math
num
pipe
pkg
rand
AbstractSmallRandom.kt
AbstractSponge.kt
AbstractSponge1024.kt
AbstractSponge256.kt
AbstractSponge512.kt
Entropy.kt
InitializationVector.kt
reg

angelos-project-aux / src / commonMain / kotlin / org.angproj.aux / rand / AbstractSponge.kt

kristoffer-paulsson PullPipe does complete basic binary piping through the sink. 28adcf2 · last year History

Code | Blame 58 lines (49 loc) · 1.72 KB Raw

```
1  /**
2   * Copyright (c) 2024 by Kristoffer Paulsson <kristoffer.paulsson@talenten.se>.
3   *
4   * This software is available under the terms of the MIT license. Parts are licensed
5   * under different terms if stated. The legal terms are attached to the LICENSE file
6   * and are made available on:
7   *
8   *      https://opensource.org/licenses/MIT
9   *
10  * SPDX-License-Identifier: MIT
11  *
12  * Contributors:
13  *      Kristoffer Paulsson - initial implementation
14  */
15 package org.angproj.aux.rand
16
17 import org.angproj.aux.util.DataBuffer
18 import org.angproj.aux.util.EndianAware
19 import org.angproj.aux.util.floorMod
20
21 /**
22  * AbstractSponge is a class that circumvents unnecessary
23  * bugs when implementing a secure random sponge construction.
24  * */
25 public abstract class AbstractSponge(spongeSize: Int = 0, public val visibleSize: Int = 0) : EndianAware {
26
27     protected var counter: Long = 0
28     protected var mask: Long = 0
29     protected val sponge: LongArray = LongArray(spongeSize) { InitializationVector.entries[it+1].iv }
30     public val byteSize: Int = visibleSize * Long.SIZE_BYTES
31
32     init {
```