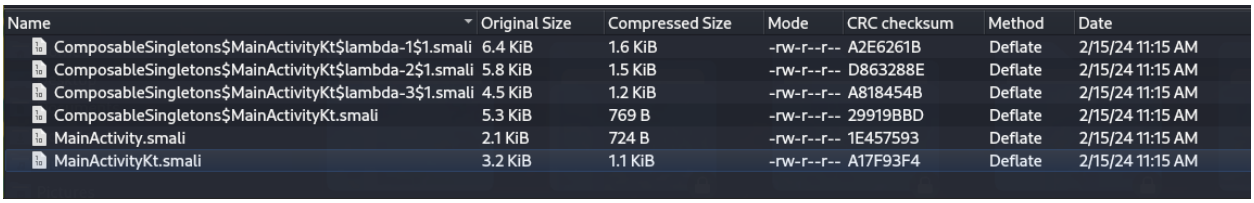
Название:	Эльфийский смали
Категория:	Реверс-инжиниринг
Уровень:	Средний
Очки:	500
Описание:	Эльфы создлали своё приложение под зелёного робота. Я смог разобрать его, но докопаться до сути - нет. Но точно одно: разобранный вариант - это какой-то смали. Хоть и всё на эльфийском.
Теги:	Android, Smali
Автор:	ROP

Прохождение:

Изучаем файлы из архива.



Name	Original Size	Compressed Size	Mode	CRC checksum	Method	Date
ComposableSingletons$MainActivityKt$lambda-1$1.smali	6.4 KiB	1.6 KiB	-rw-r--r--	A2E6261B	Deflate	2/15/24 11:15 AM
ComposableSingletons$MainActivityKt$lambda-2$1.smali	5.8 KiB	1.5 KiB	-rw-r--r--	D863288E	Deflate	2/15/24 11:15 AM
ComposableSingletons$MainActivityKt$lambda-3$1.smali	4.5 KiB	1.2 KiB	-rw-r--r--	A818454B	Deflate	2/15/24 11:15 AM
ComposableSingletons$MainActivityKt.smali	5.3 KiB	769 B	-rw-r--r--	29919BBD	Deflate	2/15/24 11:15 AM
MainActivity.smali	2.1 KiB	724 B	-rw-r--r--	1E457593	Deflate	2/15/24 11:15 AM
MainActivityKt.smali	3.2 KiB	1.1 KiB	-rw-r--r--	A17F93F4	Deflate	2/15/24 11:15 AM

Это Smali-код от Android-приложения. Сможем восстановить исходник при желании:

```
package com.codeby.smali

import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent
```

```
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.material3.MaterialTheme
import androidx.compose.material3.Surface
import androidx.compose.material3.Text
import androidx.compose.runtime.Composable
import androidx.compose.ui.Modifier
import androidx.compose.ui.tooling.preview.Preview
import com.codeby.smali.ui.theme.SmaliTheme

class MainActivity : ComponentActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContent {
            SmaliTheme {
                // A surface container using the 'background'
color from the theme
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.backgro
und
                ) {
                    // Добавление функции для преобразования
байт в строку с XOR
                    val string = byteArrayToString(byteArrayO
f(0x44,0x48,0x43,0x42,0x45,0x5e,0x7c,0x74,0x6a,0x47,0x6b,0x6
e,0x58,0x64,0x37,0x63,0x42,0x58,0x61,0x37,0x55,0x58,0x66,0x6
9,0x43,0x75,0x37,0x6e,0x63,0x7a))
                    Text(text = "-> $string")
                }
            }
        }
    }
}

fun byteArrayToString(input: ByteArray, xorKey: Byte = 0x7):
String {
```

```
    val result = StringBuilder()
    input.forEach { byte ->
        val xorResult = (byte.toInt() xor xorKey.toInt()).toB
yte()
        result.append(xorResult.toChar())
    }
    return result.toString()
}
```

А уже с ним просто поксорить каждый байт массива с 0x7.

```
0x44,0x48,0x43,0x42,0x45,0x5e,0x7c,0x74,0x6a,0x47,0x6b,0x6e,0
x58,0x64,0x37,0x63,0x42,0x58,0x61,0x37,0x55,0x58,0x66,0x69,0x
43,0x75,0x37,0x6e,0x63,0x7a
```