УДК 343.98 **DOI https://doi.org/10.32782/2522-9656/2023-13-6**

Шабля Олександр Миколайович, кандидат технічних наук, старший науковий співробітник, головний судовий експерт лабораторії досліджень об’єктів інформаційних технологій, телекомунікаційних систем та засобів Одеського науково-дослідного інституту судових експертиз Міністерства юстиції України
ORCID ID: 0000-0002-2777-2102

Старенький Іван Володимирович, судовий експерт сектору досліджень телекомунікаційних систем та засобів лабораторії досліджень об’єктів інформаційних технологій, телекомунікаційних систем та засобів Одеського науково-дослідного інституту судових експертиз Міністерства юстиції України
ORCID ID: 0009-0004-2271-8512

ДОСЛІДЖЕННЯ СЕРВІСУ ОНЛАЙН-КАЗИНО “SIMPLE GAMES” ДЛЯ ОПЕРАЦІЙНОЇ СИСТЕМИ WINDOWS

RESEARCH OF THE “SIMPLE GAMES” ONLINE CASINO SERVICE FOR THE WINDOWS OPERATING SYSTEM

Анотація. У статті розглянуто порядок та послідовність дій експерта під час дослідження програмного забезпечення “Simple Games” для операційної системи “Windows”, виявленого серед інформаційного наповнення в пам’яті побітової копії накопичувача інформації, вилученого із системного блоку, на базі якого було організовано «гральний термінал» підпільного казино. Проведено динамічний та статичний аналіз виконуючого файлу “client.exe”, що ініціалізує запуск програмного забезпечення “Simple Games”. Під час динамічного аналізу було встановлено IP-адресу та доменне ім’я WEB-ресурсу в Інтернеті, з яким під час свого виконання намагається встановити з’єднання файл “client.exe”. Під час статичного аналізу було виявлено зв’язок виконуючого файлу “client.exe” з конфігураційними файлами, що відповідають за успішну роботу програмного забезпечення “Simple Games”, що дозволило встановити логічність у послідовності дослідницьких дій експерта під час дослідження цього програмного продукту.

Ключові слова: Simple Games, казино, гральні автомати, операційна система, Windows, Linux, сервер, поштовий сервер, скрипт, спрайт, тег, файл, IP-адреса, доменне ім’я, WHOIS.

Absctract. The article considers the order and sequence of the expert’s actions in the case of researching the software “Simple Games” for the operating system “Windows” revealed among the information content in the memory of a bit-by-bit copy of the information storage, taken from the system block, on the basis of which the “grand terminal” of the underground casino was organized. A dynamic and static analysis of the executable file “client.exe”, which initializes the launch of the “Simple Games” software, was carried out. During the dynamic analysis, the IP address and domain name of the WEB resource on the Internet, with which, during its execution, the “client.exe” file tries to establish a connection, was established. During the static analysis, the connection of the executable file “client.exe” with the configuration files responsible for the successful operation of the “Simple Games” software was revealed,

which made it possible to establish logic in the sequence of the expert’s action researching this software product.

Key words: Simple Games, casino, slot machines, operating system, Windows, Linux, server, mail server, script, sprite, tag, file, IP address, domain name, WHOIS.

Вступ. На 2023 р. в Україні, згідно з інформацією Комісії з регулювання азартних ігор та лотерей (КРАІЛ), мають дозволи на гральний бізнес 57 компаній (у чотирьох із них ліцензії анульовані у 2022 р.). Більшість зареєстрована в Києві. Майже 60 % компаній працюють за основним КВЕДом «Організування азартних ігор». Закон, який легалізував гральний бізнес в Україні, набув чинності у серпні 2020 р. Комісія з азартних ігор «КРАІЛ» почала видавати перші ліцензії у лютому 2021-го. Ліцензований гральний бізнес в Україні налічує 53 компанії. На кінець березня 2023 р. нових ліцензій видано не було, тому досліджували компанії за 2021–2022 рр. 45 із них зареєстровано у Києві, 3 – у Дніпропетровській області, 2 – у Харківській, а також по одній компанії у Житомирі, Львові та Одесі. Проте місце провадження діяльності частини досліджуваних компаній поширюється не лише на регіон реєстрації, а й на інші області. Це стосується компаній, що мають дозвіл на організацію азартних ігор у залах з гральним обладнанням.

Хоча в Україні активно діє кампанія щодо боротьби з нелегальним гральним бізнесом, багато азартних закладів без ліцензії продовжують свою роботу та використовують різноманітні програмно-апаратні комплекси (ПАК) для ведення своєї діяльності. Одними з найпоширеніших програмних продуктів (ПП), що використовуються такими закладами, є сервіс “Simple Games”, що організований на базі електронно-обчислювальної машини (ЕОМ) під керуванням операційних систем (ОС) “Linux” та “Windows”.

Матеріали та метод. Матеріалами дослідження для написання цієї роботи було використано виконуючий файл (ВФ), що ініціалізує запуск ПЗ “Simple Games”, конфігураційні файли, що забезпечують роботу такого ПП, які були виявлені під час дослідження цифрового носія інформації, вилученого зі складу системного блоку персонального комп’ютера, що використовувався у нелегальному закладі казино.

Під час написання цієї роботи автори спиралися на чинні методики Міністерства юстиції України, а саме: «10.9.04», «10.9.07», «10.9.06» та «10.9.14».

Результати. Так, у [1] було проведено досліджено ПП “Simple Games”, організованого на базі ЕОМ з ОС “Linux”, а в цій роботі буде проведено дослідження такого ПЗ для ЕОМ на базі ОС “Windows”.

Більшість ПП у середовищі ОС “Windows” ініціалізуються до виконання за допомогою “*.exe”-файлу, що є зкомпільованим файлом ПП, та який компілюється в певному середовищі для програмування комп’ютерних програм, наприклад, “Visual Studio” [2].

ПЗ “Simple Games” запускається за допомогою ВФ “client.exe”, розташування якого наведено на рис. 1.

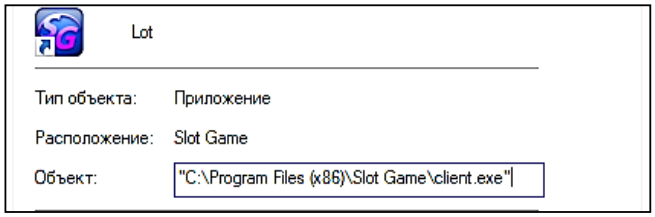


Рис. 1. Властивості “*.lnk”-файлу, що посилається на ВФ “client.exe”

Вміст теки “C:/Program Files (x86)/Slot Game/” наведено на рис. 2.

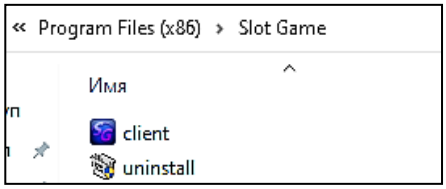


Рис. 2. Вміст теки “C:/Program Files (x86)/Slot Game/”

Своєю чергою в теці “C:/Users/<User_Name>/AppData/Local/Slot Game/main”, вміст якої наведено на рис. 3, знаходяться конфігураційні файли до ПЗ “Simple Games”, що запускається виконуючим файлом “client.exe”, на який посилається файл “Lot.lnk”.

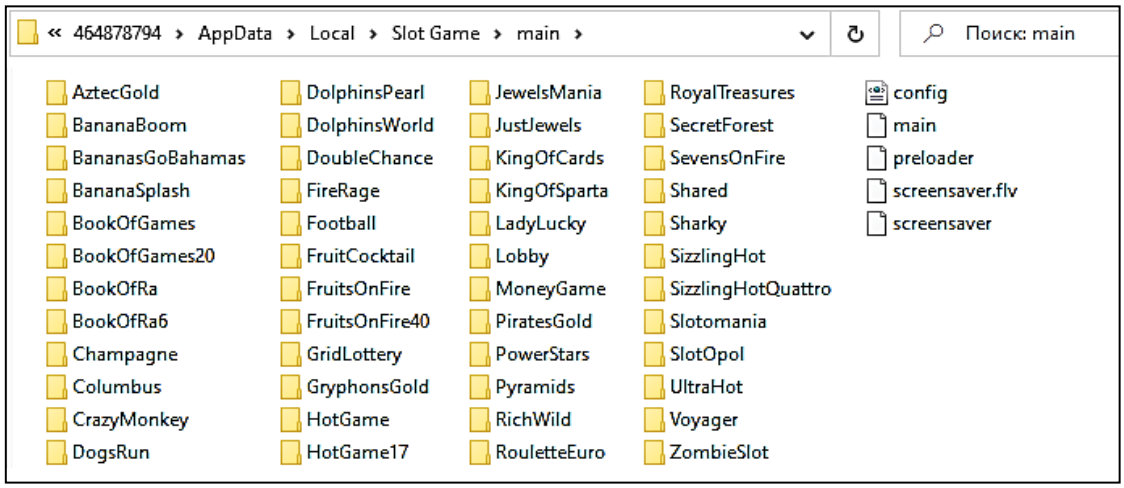


Рис. 3. Вміст теки “C:/Users/<User_Name>/AppData/Local/Slot Game/main”

Загалом дослідження будь-якого виконуючого файлу (ВФ) можна поділити на два етапи дослідження – динамічний та статичний.

Тому під час динамічного дослідження файлу “client.exe” проводиться його безпосередній запуск (в ізольованому середовищі, у якості якого може бути використана спеціально для цього створена віртуальна машина). Стартове вікно ПЗ “Simple Games”, що відкривається користувачеві після виконання файлу “client.exe”, наведено на рис. 4.

Як видно з рис. 4, стартове вікно виконуючого файлу “client.exe” містить поле для введення 16-значного цифрового паролю “Enter ticket number:”, що з великою долею вірогідності виконує функцію персоніфікації користувача.

Без діючого актуального паролю ПАК “Simple Games” здійснити вхід до середовища ПЗ “Simple Games” та продовжити динамічний аналіз файлу “client.exe” неможливо.

Рис. 4. Стартове вікно виконуючого файлу “client.exe”

Оскільки ПЗ “Simple Games” позиціонується як ПП, що дає можливість реалізувати функції грального автоматів, відеоатракціонів, лотерейних терміналів та конструктивно схожих з ними пристроїв – електронного (віртуального) казино, слот-«машин», для його успішної роботи потрібен постійний доступ до Інтернету. А отже, з точки зору динамічного аналізу файлу “client.exe” є доцільним відстеження WEB-ресурсів в Інтернеті, до яких звертається файл “client.exe”.

За допомогою ПЗ “Microsoft Network Monitor 3.4” [3] було встановлено IP-адресу та доменне ім’я WEB-ресурсу, з яким ВФ “client.exe” намагається встановити Інтернет-з’єднання, у спробі введення довільної комбінації 16-значного цифрового паролю у вікно “Enter ticket number:” (див. рис. 5).

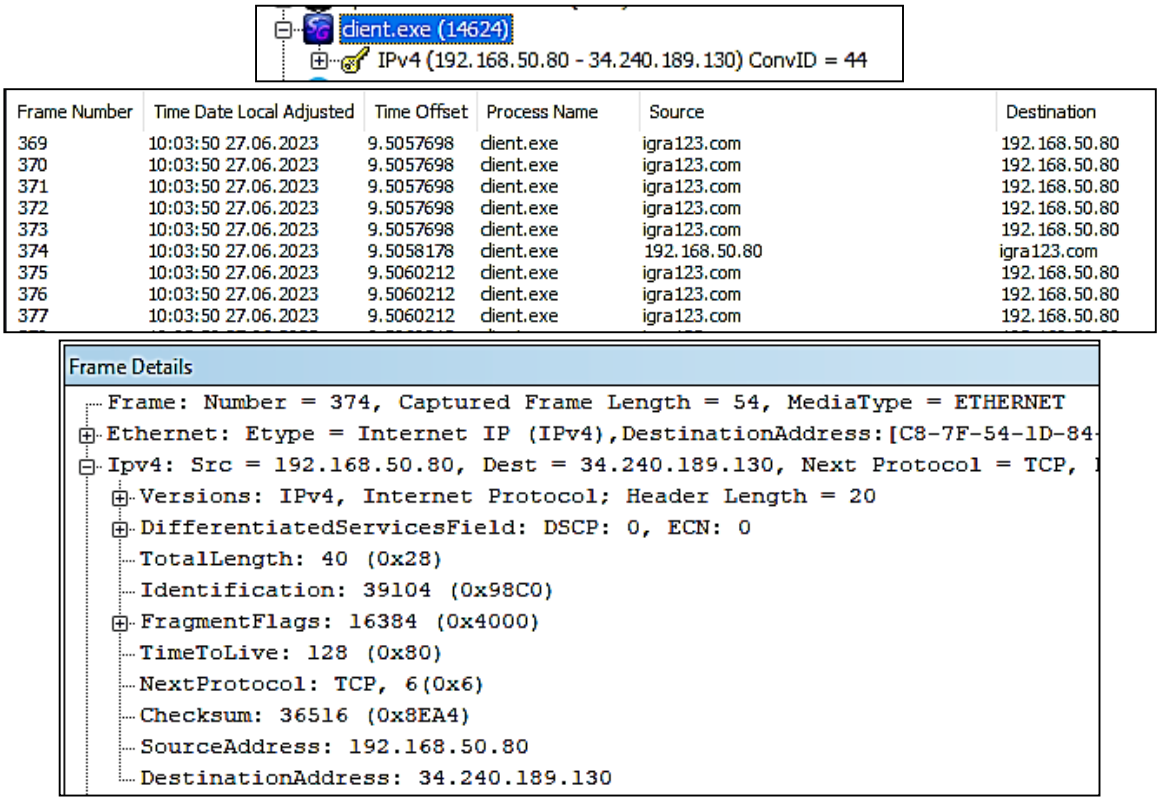


Рис. 5. Відстеження Інтернет-трафіку ПЗ “client.exe” за допомогою ПЗ “Microsoft Network Monitor 3.4”

На момент написання цієї роботи за допомогою сервісу “https://2ip.ua/” в глобальній мережі Інтернет було встановлено WHOIS-інформацію щодо WEB-ресурсу “igra123.com”:

Ім'я провайдера:	Amazon.com Inc.
Сайт провайдера:	http://www.amazon.com/
Номер AS провайдера:	16509
Місцезнаходження:	Ірландія, Дублін
Інформація про домен:	Дізнатися
DNS параметри домена:	Дізнатися
Хост, що перевіряється:	ec2-34-240-189-130.eu-west-1.compute.amazonaws.com (34.240.189.130)
Доступність хоста:	Перевірити
Активні сервіси:	Перевірити
NetRange:	34.240.0.0 - 34.247.255.255
CIDR:	34.240.0.0/13
NetName:	AMAZON-DUB
NetHandle:	NET-34-240-0-0-1
Parent:	AT-88-Z (NET-34-192-0-0-1)
NetType:	Reallocated
OriginAS:	AS16509
Organization:	Amazon Data Services Ireland Limited (ADSIL-1)
RegDate:	2017-05-18
Updated:	2017-05-18
Ref:	34.240.0.0" rel="nofollow" target="_blank">https://rdap.arin.net/registry/ip/34.240.0.0
OrgName:	Amazon Data Services Ireland Limited
OrgId:	ADSIL-1
Address:	Unit 4033, Citywest Avenue Citywest Business Park
City:	Dublin
StateProv:	D24
PostalCode:	
Country:	IE
RegDate:	2014-07-18
Updated:	2014-07-18
Ref:	https://rdap.arin.net/registry/entity/ADSIL-1
OrgTechHandle:	ANO24-ARIN
OrgTechName:	Amazon EC2 Network Operations
OrgTechPhone:	+1-206-555-0000
OrgTechEmail:	amzn-noc-contact@amazon.com
OrgTechRef:	https://rdap.arin.net/registry/entity/ANO24-ARIN
OrgAbuseHandle:	AEA8-ARIN
OrgAbuseName:	Amazon EC2 Abuse
OrgAbusePhone:	+1-206-555-0000
OrgAbuseEmail:	abuse@amazonaws.com
OrgAbuseRef:	https://rdap.arin.net/registry/entity/AEA8-ARIN
OrgNOCHandle:	AANO1-ARIN
OrgNOCName:	Amazon AWS Network Operations
OrgNOCPhone:	+1-206-555-0000
OrgNOCEmail:	amzn-noc-contact@amazon.com
OrgNOCRef:	https://rdap.arin.net/registry/entity/AANO1-ARIN

За допомогою команди терміналу ОС “Linux” ***host*** можна встановити перелік IP-адрес серверів, що обслуговують доменне ім’я “igra123.com”, а також встановити свідчення щодо поштового серверу, що закріплений за цим доменним іменем.

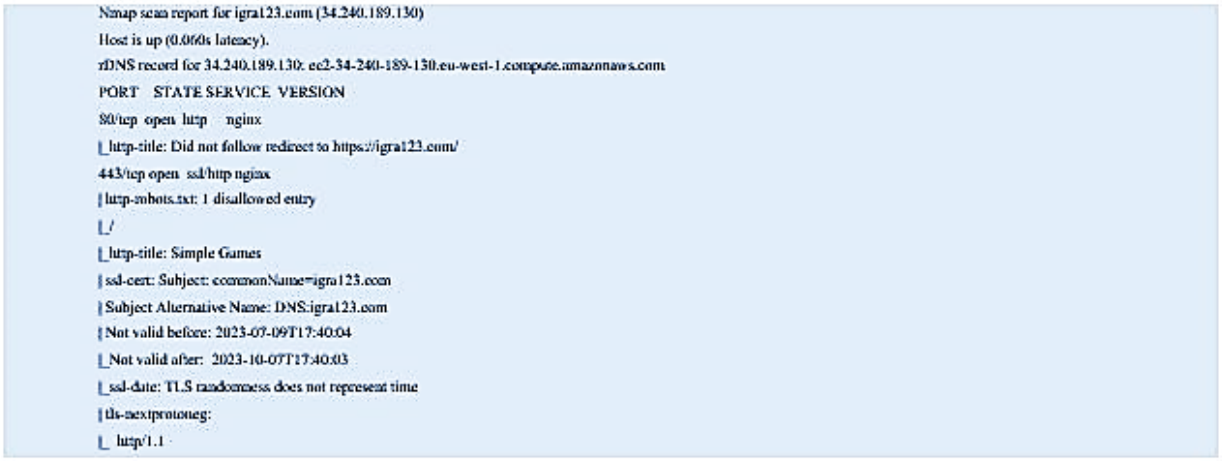
Нижче наведено результат виконання команди ***host*** щодо WEB-ресурсу “igra123.com”.

```
$ host igra123.com
igra123.com has address 34.240.189.130
igra123.com mail is handled by 10 mail.igra123.org.
```

За допомогою команди терміналу ОС “Linux” ***nmap*** можна просканувати WEB-ресурс “igra123.com” на предмет відкритих “http” та “https” портів,

встановити тип поштового сервісу, що встановлений на серверному обладнанні, а також встановити заголовок головної сторінки цього WEB-ресурсу.

Результат команди ***nmap*** щодо WEB-ресурсу “igra123.com” наведено нижче.



Так, за допомогою динамічного аналізу файлу “client.exe” було встановлено IP-адресу та доменне ім’я WEB-ресурсу (“igra123.com”), до якого під час свого виконання звертається останній.

Своєю чергою за допомогою двох команд терміналу ОС “Linux” було встановлено загальну кількість серверів, що надають доступ до цього WEB-ресурсу, адресу поштового серверу, тип такого серверу (NGINX), а також заголовок головної сторінки WEB-ресурсу “igra123.com” – “Simple Games”.

Статичний аналіз зкомпільованих виконуючих файлів зручно проводити за допомогою спеціалізованого ПЗ для проведення зворотного програмування (reverse engineering) ЗП.

Так, за допомогою ПЗ “Ghidra” [4] було проведено процес зворотного програмування виконуючого файлу “client.exe”.

Під час дослідження результатів сканування файлу “client.exe” виявлено свідчення посилання на файли “main.swf”, “screensaver.swf”, а також файл “config.ini”, в одній з функцій файлу (функція “FUN_00403a50”) “client.exe” (див. рис. 6).

```
iVar4 = FUN_004096f0((HWND *)0x0);
pcVar7 = "screensaver.swf";
if (iVar4 == 0) {
  pcVar7 = "main.swf";
}
_memset(local_e4,0,200);
iVar4 = FUN_00409360();
if (iVar4 != 0) {
  pCVar6 = FUN_004094a0((LPCSTR)0x0,"config.ini");
  GetPrivateProfileStringA
```

Рис. 6. Вміст функції “FUN_00403a50” виконуючого файлу “client.exe”

На рис. 7 наведено вміст функції “FUN_00402910” виконуючого файлу “client.exe”, що відповідає за встановлення Інтернет-з’єднання з WEB-ресурсом “igra123.org”.

```
undefined4 FUN_00402910(void)

{
  PCNZCH pCVar1;

  pCVar1 = (PCNZCH)FUN_00408850();
  pCVar1 = FUN_00409420(pCVar1,"igra123.org");
  if (pCVar1 != (PCNZCH)0x0) {
    MessageBoxA(hWnd_0051bca8,"Exception","Error",0x10);
  }
  return 0;
```

Рис. 7. Вміст функції “FUN_00402910” виконуючого файлу “client.exe”

Як уже було встановлено раніше, адреса “mail.igra123.org” є адресою поштового серверу WEB-ресурсу “igra123.com”, про що свідчить результат команди host щодо WEB-ресурсу “igra123.org”, який наведено нижче:

```
$ host igra123.org
igra123.org has address 85.10.216.176
igra123.org mail is handled by 10 mail.igra123.org
```

На рис. 8 наведено вміст функції “FUN_00408850” виконуючого файлу “client.exe”, в тексті якої виявлено функцію, що взаємодіє з файлом “config.ini” (функція “FUN_004094a0”), а також міститься рядок з виведення інформаційного повідомлення, що містить словосполучення “Simple Touch” та “Slot Game”, а також міститься рядок “GetPrivateProfileString”, що відповідає за роботу з файлами “*.ini”.

```
if ((char)lpData_0051dbe8 != '\0') goto LAB_00408ac0;
_memset(&lpData_0051dbe8,0,0x100);
wsprintfA(local_218,"Software\\Simple Touch\\%s","Slot Game");
local_220 = (HKEY)0x0;
dwErrCode = RegOpenKeyExA((HKEY)0x80000001,local_218,0,0x20019,&local_220);
SetLastError(dwErrCode);
if (local_220 == (HKEY)0x0) {
LAB_00408a5b:
  _memset(&lpData_0051dbe8,0,0x100);
  pCVar4 = FUN_004094a0((LPCSTR)0x0,"config.ini");
  GetPrivateProfileStringA
            ("General","Server",(LPCSTR)&lpNewItem_004f60fc,(LPSTR)&lpData_0051dbe8,0x100,pCVar4);
}
else {
  local_21c = 0xff;
  LVar6 = RegQueryValueExA(local_220,"server",(LPDWORD)0x0,(LPDWORD)0x0,(LPBYTE)&lpData_0051dbe8,
                           &local_21c);
```

Рис. 8. Вміст функції “FUN_00408850” виконуючого файлу “client.exe”

Як було встановлено раніше, для своєї роботи файл “client.exe” використовує “*.swf” файли.

У теці “C:/Users/<User_Name>/AppData/Local/Slot Game/main” виявлено файл “config.xml”, в якому прописано основні конфігураційні моменти для роботи файлу “client.exe”. Так, у разі проведення зворотного програмування файлу “client.exe” було виявлено функції, що відповідають за взаємодію з файлом “config.xml”. Частковий вміст однієї з таких функцій наведено на рис. 9.

```
17    pCVar2 = FUN_004094a0((LPCSTR)0x0,"config.ini");
18    GetPrivateProfileStringA
19              ("General",(LPCSTR)&lpKeyName_004f5a6c,(LPCSTR)&lpString2_004f5a68,local_190,200,
20               pCVar2);
21  }
22  if (local_190[0] != '\0') {
23    iVar1 = CompareStringA(0x800,0x1003,local_190,-1,(PCNZCH)&lpString2_004f5a68,-1);
24    pCVar2 = local_190;
25    if (iVar1 != 2) goto LAB_00406786;
26  }
27  pCVar2 = "main";
28LAB_00406786:
29  pCVar2 = FUN_004094a0(pCVar2,"config.xml");
30  lstrcpyA((LPSTR)&lpFile_0051ed28,pCVar2);
31  this = FUN_00409fc0((LPCSTR)&lpFile_0051ed28);
32  if (this != (HLOCAL)0x0) {
33    FUN_004065c0(this,(int)local_c8);
34    FUN_00406490(param_1,"flash_version",(int)local_68);
```

Рис. 9. Частковий вміст функції “FUN_004066e0” виконуючого файлу “client.exe”

Файл “config.xml” містить у собі 446 рядків інформаційного наповнення. Рядки з 1 по 41 містять у собі інформацію щодо вибору мови інтерфейсу ПЗ, вибору типу валюти, а також опис головного екрана (тег <home>) та навколишнього середовища ПЗ (тег < gridlottery >).

Інформаційний вміст таких рядків наведено нижче:

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <params name="common">
    <param name="flashVersion" value="2.325"/>
  </params>
  <languages>
    <lang id="en" name="English" label="EN"/>
    <lang id="ua-L" name="Українська" label="UA"/>
    <lang id="ua" name="Українська" label="UA"/>
    <lang id="ru" name="Русский" label="RU"/>
    <lang id="de" name="Deutsch" label="DE"/>
    <lang id="fr" name="Français" label="FR"/>
    <lang id="es" name="Español" label="ES"/>
    <lang id="pl" name="Polski" label="PL"/>
    <lang id="ja" name="日本語" label="JA"/>
    <lang id="zh" name="中文" label="ZH"/>
    <lang id="ka" name="ქართული" label="KA"/>
    <lang id="el" name="Ελληνικά" label="EL"/>
    <lang id="bg" name="Български" label="BG"/>
    <lang id="ro" name="Română" label="RO"/>
    <lang id="cs" name="Čeština" label="CS"/>
    <lang id="pt" name="Português" label="PT"/>
  </languages>
  <currency>
    <item id="UAH" name="Гривня" label="₴"/>
    <item id="USD" name="US Dollar" label="$"/>
    <item id="EUR" name="Euro" label="€"/>
    <item id="RUB" name="Рубль" label="₽"/>
  </currency>
  <files>
    <home>
      <file src="Lobby/Lobby.swf" type="main" tm="1466062166"/>
      <file src="Shared/jackpot_graphics.swf" name="jackpots" tm="v2"/>
    </home>
    <gridlottery>
      <file src="GridLottery/game.swf" type="main" tm="1466062161"/>
      <file src="GridLottery/graphics.swf" name="graphics" tm="1465886161"/>
      <file src="GridLottery/sounds.swf" name="sounds" tm="1465833276"/>
      <file src="Shared/jackpot_graphics.swf" name="jackpots" tm="v2"/>
      <file src="Shared/voucher.swf" name="voucher" tm="v1"/>
    </gridlottery>
```

Всі наступні рядки файлу “config.xml” містять опис кожної з ігор (тег <game>), що доступні у середовищі ПЗ “client.exe”.

Нижче, для прикладу, наведено вміст опису гри “aztecGold”.

```
<game id="16" name="aztecGold">
    <file src="AztecGold/game.swf" type="main" tm="1465888099"/>
    <file src="AztecGold/graphics.swf" name="graphics" tm="1465886160"/>
    <file src="AztecGold/sounds.swf" name="sounds" tm="1465833276"/>
    <file src="AztecGold/symbols.swf" name="symbols" tm="1465806695"/>
    <file src="Shared/jackpot_anim.swf" name="jpAnim" type="dynamic" tm="v2"/>
    <file src="Shared/jackpot_graphics.swf" name="jackpots" tm="v2"/>
    <file src="Shared/voucher.swf" name="voucher" tm="v1"/>
</game>
```

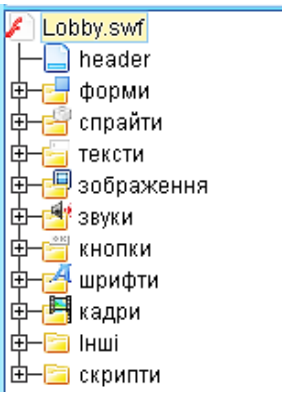


Рис. 10. «Кореневa структура» файлу “Lobby.swf”

Як видно з наведеного вмісту файлу “config.xml”, наступним після опціонального опису вибору мови інтерфейсу та валюти у середовищі ПЗ є звернення до файлу “Lobby.swf”.

Так, на рис. 10 наведено «кореневу структуру» файлу “Lobby.swf”, виявленого в теці “C:/Users/<User_Name>/AppData/Local/Slot Game/main/ Lobby”.

Розглянемо більш детально вміст кожного з елементів кореневої структури файлу “Lobby.swf”.

На рис. 11 наведено частковий вміст вкладки «форми».

На рис. 12 наведено частковий вміст вкладки «спрайти».

Рис. 11. Частковий вміст вкладки «форми»

Рис. 11. Закінчення

Рис. 12. Частковий вміст вкладки «спрайти»

На рис. 13 наведено частковий вміст вкладки «тексти».

На рис. 14 наведено частковий вміст вкладки «зображення», а на рис. 15 наведено зовнішній вигляд зображення “DefineBitsJPEG3 (202)”.

На рис. 16 наведено вміст вкладки «шрифти».

Вміст вкладки «скрипти» наведено на рис. 17.

Як видно з рис. 17, файл “Lobby.swf” містить у собі 16 пакетів, у кожному з яких містяться скрипти (сценарії), що забезпечують роботу файлу “Lobby.swf” у загальному «комплексі» ПЗ “Simple Games” у разі виконання файлу “Client.exe”.

Так, наприклад, пакет “lobby” містить скрипт “Config”, в якому наведено налаштування розмірів робочого вікна, отримання інформації щодо ідентифікатора терміналу (робочої станції), номер білета (TICKET_CODE), тип валюти та ін.

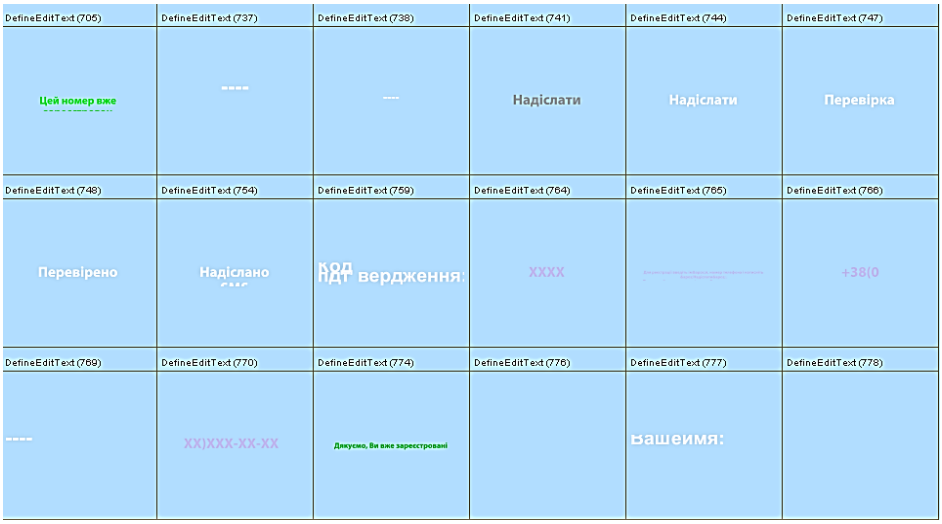


Рис. 13. Частковий вміст вкладки «тексти»

Рис. 14. Частковий вміст вкладки «зображення»

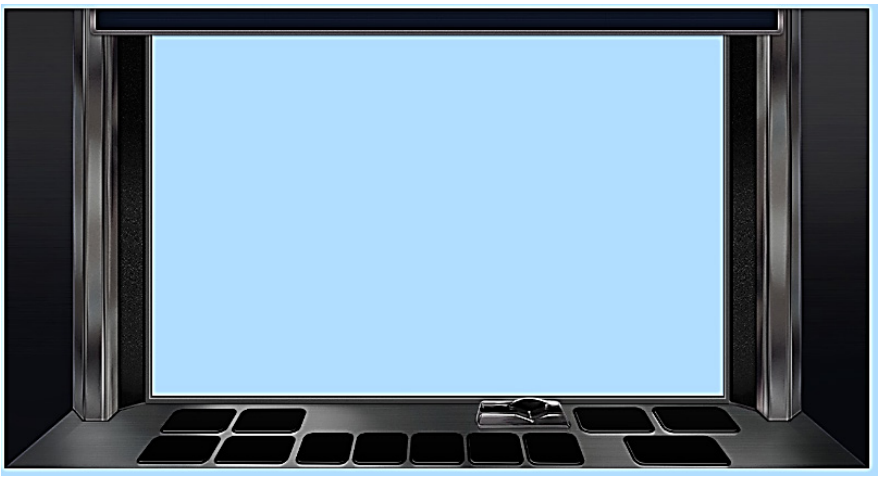


Рис. 15. Зовнішній вигляд зображення “DefineBitsJPEG3 (202)”

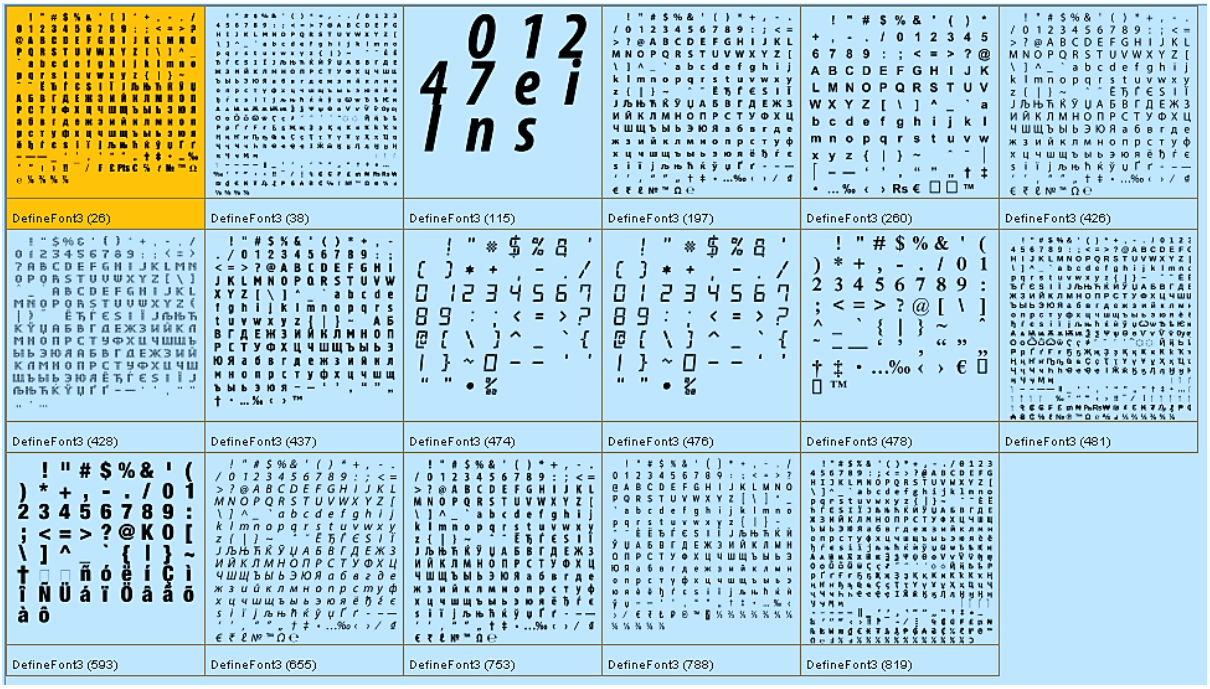


Рис. 16. Вміст вкладки «шрифти»

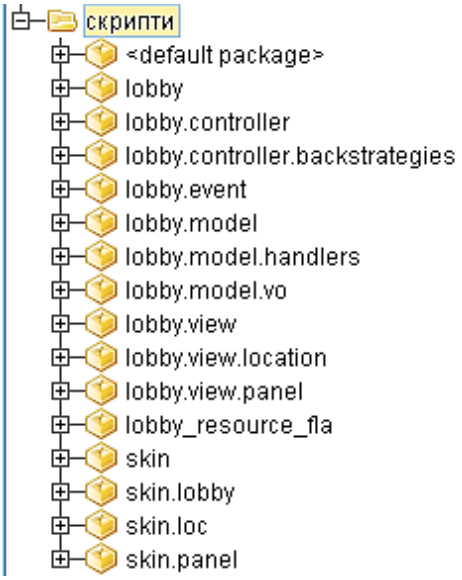


Рис. 17. Вміст вкладки «скрипти»

Вміст скрипту “Config” пакета “lobby” файлу “Lobby.swf” наведено нижче:

```
package lobby
{
  public class Config
  {

    public static const APP_WIDTH:uint = 1024;
    public static const APP_HEIGHT:uint = 672;
    public static const MODULE_NAME:String = "lobby";
    public static var isMatchVersions:Boolean = true;
    private static var _TERMINAL_ID:String = "";
    private static var _TICKET_CODE:String = "";
    private static var _FLASH_VERSION:String = "";
    private static var _APP_VERSION:String = "";
    private static var _IP1:String = "";
    private static var _IP2:String = "";
    private static var _HASH:String = "";
    private static var _HARDWARE_ID:String = "";
    private static var _CLIENT_TYPE:String = "";
    private static var getCurrency:Function;
    private static var _CURRENCY:String = "";

    public function Config()
    {
      super();
    }
    public static function get TERMINAL_ID() : String
    {
      return _TERMINAL_ID;
    }
    public static function get TICKET_CODE() : String
    {
      return _TICKET_CODE;
    }

    public static function get FLASH_VERSION() : String
    {
      return _FLASH_VERSION;
    }
    public static function get APP_VERSION() : String
    {
      return _APP_VERSION;
    }
    public static function get IP1() : String
    {
      return _IP1;
    }
    public static function get IP2() : String
    {
      return _IP2;
    }
    public static function get HASH() : String
    {
      return _HASH;
    }
    public static function get HARDWARE_ID() : String
    {
      return _HARDWARE_ID;
    }
    public static function get CLIENT_TYPE() : String
    {
      return _CLIENT_TYPE;
    }
    public static function get CURRENCY() : String
    {
      return _CURRENCY;
    }
    public static function initCurrency(param1:String) : void
    {
      if(getCurrency is Function)
      {
        _CURRENCY = getCurrency(param1);
      }
    }
    public static function parseGlobalParams(param1:Object) : void
    {
      _TERMINAL_ID = !!param1["terminal"] ? String(param1["terminal"]) : "";
      _FLASH_VERSION = param1["flashVersion"];
      _APP_VERSION = param1["appVersion"];
      _HARDWARE_ID = param1["hardwareID"];
      _IP1 = param1["IP1"];
      _IP2 = param1["IP2"];
      _HASH = param1["hash"];
      _TICKET_CODE = param1["code"];
      _CLIENT_TYPE = param1["clientType"];
      isMatchVersions = param1["isMatchVersions"];
      if(param1.hasOwnProperty("getCurrency"))
      {
        getCurrency = param1["getCurrency"] as Function;
      }
    }
  }
}
```

Під час дослідження скриптів, що містяться в файлі “main.swf”, звернення до якого було виявлено при ЗП файлу “client.exe”, в пакеті “mainapp”, було виявлено скрипт “Config”, який відповідає за взаємодію із зовнішнім WEB-ресурсом у Інтернеті. Вміст такого скрипту наведено нижче:

```
package mainapp
{
  import flash.events.Event;
  import flash.external.ExternalInterface;
  import flash.system.Capabilities;
  import mainapp.model.MainModel;
  import mainapp.view.DisplayLog;
  import shared.settings.SettingKey;
  import shared.settings.SettingsRepository;
  import shared.utils.Log;
  import shared.utils.ValidURL;

  public class Config
  {
    public static const APP_WIDTH:uint = 1024;
    public static const APP_HEIGHT:uint = 672;
    public static var isRatioProportional:Boolean = false;
    public static var currentGameID:String = "1";
    public static var FLASH_VERSION:String = "1";
    public static var APP_VERSION:String = "1";
    public static var isMatchVersions:Boolean = true;
    public static var LANGUAGES_PATH:String = "";
    public static var GUID:String = "";
    public static var WEB_PARTNER_MODE:Boolean = false;
    public static var DEFAULT_DENOM:int = 10;
    public static var SCRSAVER_SRC:String = "";
    public static var SCRSAVER_DELAY_MINUTES:uint = 20;
    private static var _instance:mainapp.Config;
    private var _callback:Function;
    private var _configLoader:mainapp.ConfigLoader;
    public function Config()
    {
      super();
      if(_instance)
      {
        throw new Error("Error: Instantiation failed: Use Config.instance instead of new.");
      }
      this._configLoader = new mainapp.ConfigLoader();
      this._configLoader.addEventListener(mainapp.ConfigLoader.XML_CONFIG_LOADED_EVENT,this.configLoader_xmlConfigLoadedHandler);
      this._configLoader.addEventListener(mainapp.ConfigLoader.JSON_CONFIG_LOADED_EVENT,this.configLoader_jsonConfigLoadedHandler);
      this._configLoader.addEventListener(mainapp.ConfigLoader.COMPLETE_LOADED_EVENT,this.configLoader_completeLoadedHandler);
    }
    public static function get instance() : mainapp.Config
    {
      if(_instance == null)
      {
        _instance = new mainapp.Config();
      }
      return _instance;
    }
    public static function get APP_DIR() : String
    {
      return SettingsRepository.instance.getBy(SettingKey.DIR);
    }
    public static function get TICKET_CODE() : String
    {
      return SettingsRepository.instance.getBy(SettingKey.CODE);
    }
    public static function get CLIENT_TYPE() : String
    {
      return SettingsRepository.instance.getBy(SettingKey.CLIENT);
    }
    public static function get LANG_LIST() : Array
    {
      var _loc1_:* = SettingsRepository.instance.getBy(SettingKey.LANG);
      if(_loc1_ == "")
      {
        return [];
      }
```

```
    return String(_loc1_).split(",");
  }
  public static function get IS_USE_RIGHT_MOUSE() : Boolean
  {
    return SettingsRepository.instance.getBy(SettingKey.IS_USE_RIGHT_MOUSE_CLICK);
  }
  public static function get POS() : String
  {
    return SettingsRepository.instance.getBy(SettingKey.POS);
  }
  public static function get HOME_FILES() : XMLList
  {
    return SettingsRepository.instance.getBy(SettingKey.HOME_FILES);
  }
  public static function get LANGUAGES() : XMLList
  {
    return SettingsRepository.instance.getBy(SettingKey.LANGUAGES);
  }
  public static function get GAME_FILES() : XMLList
  {
    return SettingsRepository.instance.getBy(SettingKey.GAME_FILES,currentGameID);
  }
  public static function get GRIDLOTTERY_FILES() : XMLList
  {
    return SettingsRepository.instance.getBy(SettingKey.GRIDLOTTERY_FILES);
  }
  public static function get LOG_URL() : String
  {
    var _loc1_:ValidURL = SettingsRepository.instance.domain;
    return _loc1_.getHttpOrHttps("/feedback.php");
  }
  public static function getCurrency(param1:String) : String
  {
    param1 ||= "";
    if(param1 == "")
    {
      return "";
    }
    return SettingsRepository.instance.getBy(SettingKey.CURRENCY,param1);
  }
  public function configure(param1:Object, param2:Function) : void
  {
    var date:Date;
    var url:ValidURL;
    var jsonURI:String;
    var params:Object;
    var fv:Object = param1;
    var cb:Function = param2;
    DisplayLog.instance.write("2.1 fv" + fv + ";");
    try
    {
      SettingsRepository.instance.flashVars = fv;
    }
    catch(e:Error)
    {
      DisplayLog.instance.write("2.1.error," + e.errorID);
    }
    APP_VERSION = SettingsRepository.instance.getBy(SettingKey.APP_VERSION) || "";
    DisplayLog.instance.write("2.2");
    this._callback = cb || new Function();
    this._configLoader.initialize();
    DisplayLog.instance.write("2.3");
    date = new Date();
    url = SettingsRepository.instance.domain;
    jsonURI = url.getHttpOrHttps("/client/config?server=" + url.domainName + "&pos_id=" + POS);
    params = {
      "jsonURI":jsonURI,
      "xmlURI":APP_DIR + "config.xml?ac=" + String(date.time)
    };
    DisplayLog.instance.write("2.4");
    this._configLoader.load(params);
    DisplayLog.instance.write("2.5");
  }
```

```
public function toString() : String
{
  return "Config{client_ver=" + FLASH_VERSION + ", flash_ver=" + Capabilities.version + ", debug=" + Capabilities.isDebugger + "}";
}
private function configLoader_xmlConfigLoadedHandler(param1:Event) : void
{
  DisplayLog.instance.write("2.6");
  SettingsRepository.instance.xmlData = this._configLoader.xmlData;
  DisplayLog.instance.write("2.7");
  FLASH_VERSION = SettingsRepository.instance.getBy(SettingKey.FLASH_VERSION) || "";
  Log.write("[INFO] " + this.toString());
  SCRSAVER_SRC = SettingsRepository.instance.getBy(SettingKey.SCREENSAVER_SOURCE);
  SCRSAVER_DELAY_MINUTES = parseInt(SettingsRepository.instance.getBy(SettingKey.SCREENSAVER_DELAY));
  LANGUAGES_PATH = APP_DIR + SettingsRepository.instance.getBy(SettingKey.LANGUAGES_PATH);
  DisplayLog.instance.write("2.8");
}
private function configLoader_jsonConfigLoadedHandler(param1:Event) : void
{
  DisplayLog.instance.write("2.9");
  SettingsRepository.instance.webData = this._configLoader.jsonData;
}
private function configLoader_completeLoadedHandler(param1:Event) : void
{
  DisplayLog.instance.write("2.10");
  this._configLoader.dispose();
  var _loc2_:* = SettingsRepository.instance.getBy(SettingKey.CODE);
  var _loc3_:* = SettingsRepository.instance.getBy(SettingKey.GAME_ID);
  if(_loc2_ != "" && _loc3_ != "")
  {
    WEB_PARTNER_MODE = true;
    currentGameID = _loc3_;
    MainModel.instance.ticketCode = _loc2_;
  }
  if(!ExternalInterface.available)
  {
    if(FLASH_VERSION != SettingsRepository.instance.getBy(SettingKey.ACTUAL_FLASH_VERSION))
    {
      isMatchVersions = false;
      Log.write("[WARN] Flash-app version mismatch");
    }
    if(APP_VERSION != SettingsRepository.instance.getBy(SettingKey.ACTUAL_APP_VERSION))
    {
      isMatchVersions = false;
      Log.write("[WARN] Exe-app version mismatch");
    }
  }
  DisplayLog.instance.write("2.11");
  this._callback();
}
}
}
```

Під час дослідження “*.swf” файлів у теці “AztecGold” (яка була вибрана довільним шляхом серед інших тек) серед інформаційного вмісту файлу “graphics.swf” на вкладках «спрайти» та «зображення» виявлено такі елементи, що наведені на рис. 18 та 19.

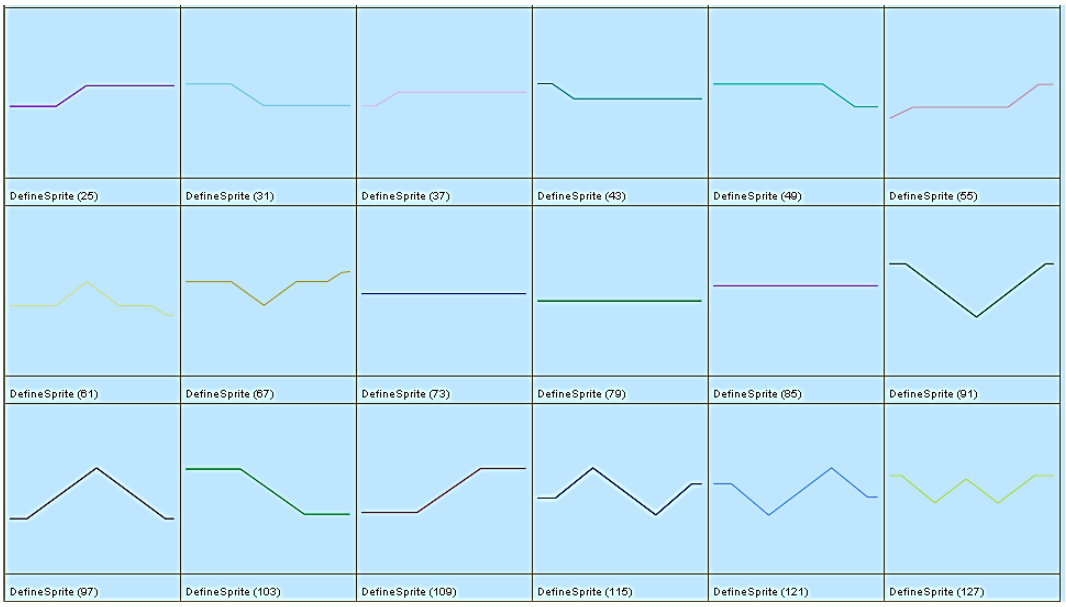


Рис. 18. Частковий вміст вкладки «спрайти»

Рис. 18. Закінчення

Серед скриптів пакета “aztecGold.controller” файлу “game.swf”, що розміщується в теці “AztecGold”, виявлено скрипт “MainController”, який містить опис у вигляді інформаційних повідомлень на певні дії користувача.

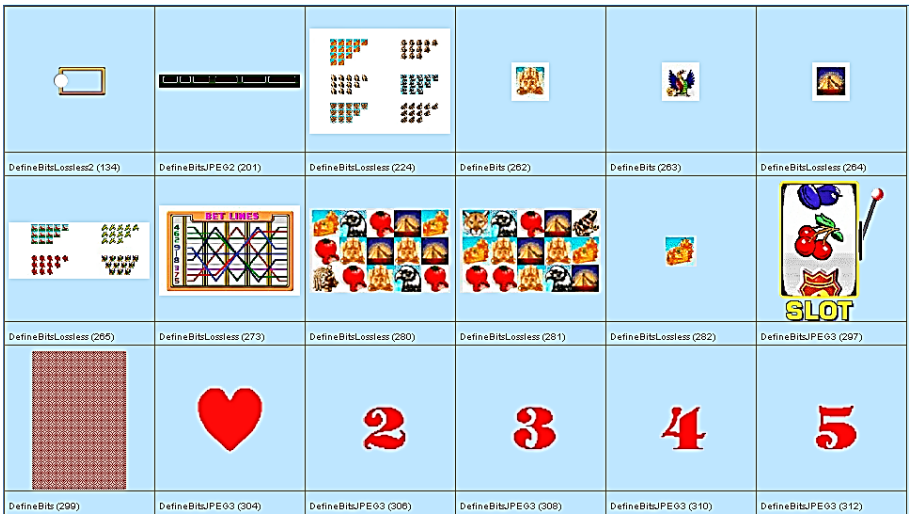


Рис. 19. Частковий вміст вкладки «зображення»

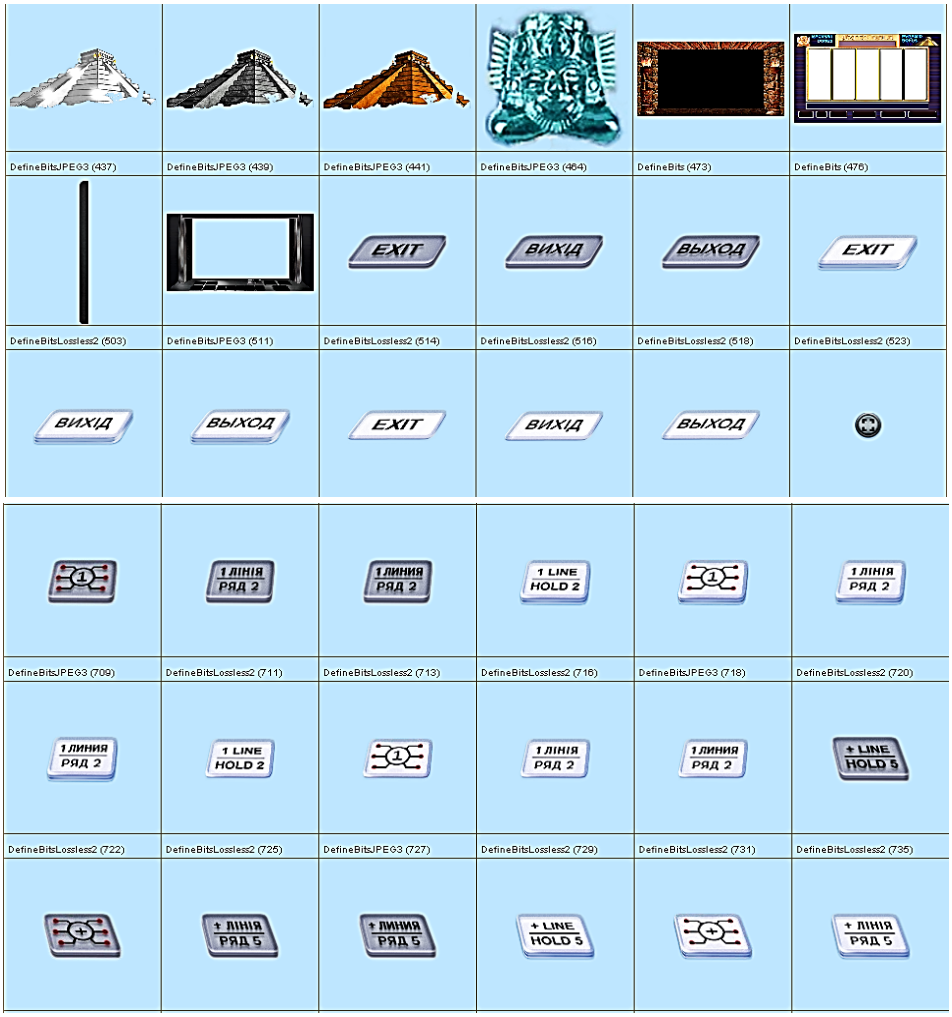


Рис. 19. Закінчення

Так, нижче наведено фрагменти скрипту “MainController”:

```
private function runNormalMode() : void
    {
      this.Model.updateBalance();
      this.Model.initDefaultBets();
      this.selectBet();
      this.panelScr.writeMessage(Language.getText("placeToBet"));
    }

public function openGamble(param1:uint = 0, param2:String = "", param3:Array = null) : void
    {
      Sounds.stopSound("dnub_waiting");
      Flag.isDoubleAllowed = false;
      Flag.isDoubleOpened = true;
      this.panel.setBtnEnabled(true,["BR","MB"]);
      this.gambleMJ = new GambleMegajack(this.closeGamble,param1,param2,param3);
      MainView.instance.openGameWindow(this.gambleMJ,"paytable");
      this.panelMsg(Language.getText("selectCard"));
      MainModel.instance.dispatchOpenPanelWindow();
      MainModel.instance.currentGame.isDoubAllowed = false;
    }
}
      aztecGold.controller.MainController.instance.panelMsg(Language.getText("gambleEnd"));
    }
public function checkMoney() : Boolean
    {
      if(MainModel.instance.isGameOver)
      {
        this.panel.setBtnEnabled(true,["DN","MN","PT"]);
        this.panelMsg(Language.getText("noMoney"));
        if(Flag.autoPlayMode)
        {
          this.panel.doStopAuto();
        }
        return false;
      }
      return true;
    }
```

З проведеного статичного дослідження файлу “client.exe” було встановлено безпосередній зв’язок з файловим вмістом теки “C:/Users/<User_Name>/AppData/Local/Slot Game/main”, що дало можливість створити логічну ланку в подальшому дослідженні файлів такої теки.

Висновки. У цій роботі було розглянуто порядок та послідовність дій експерта під час дослідження ПЗ “Simple Games” для ОС “Windows” шляхом динамічного та статичного дослідження ВФ “client.exe”, запуск якого ініціалізує виконання зазначеного програмного забезпечення.

Протягом усього дослідження відстежується послідовність логічних дій експерта під час дослідження файлу “client.exe” та файлового наповнення теки “C:/Users/<User_Name>/AppData/Local/Slot Game/main”.

Отримані результати дослідження, без активного авторизаційного коду до середовища ПЗ “Simple Games”, дозволяють зробити висновок щодо вірогідної причетності такого ПП до категорії тих, що дають можливості з реалізації на базі ЕОМ функцій гральних автоматів, відеоатракціонів, лотерейних терміналів та конструктивно схожих на них пристроїв – електронного (віртуального) казино, «слот-машини». А з наявним авторизаційним кодом є велика вірогідність продовжити динамічний аналіз та отримати доказову базу для категоричного висновку.

Перелік використаних джерел:

1. Старенький І. В., Донченко О. І. Дослідження програмного забезпечення “Simple Games” та “Iconnect” для операційної системи “Linux”. *Вісник ОНДІСЕ.* Науково-практичне видання. ISSN 2522-9656, Випуск 12. Одеса, 2022. С. 74–93.

2. Visual Studio. *Вікіпедія: вільна енциклопедія*. URL: https://en.wikipedia.org/wiki/Visual_Studio (дата звернення: 07.07.2023).

3. Microsoft Network Monitor 3.4. URL: https://www.microsoft.com/en-us/download/4865 (дата звернення: 07.07.2023).

4. Ghidra. URL: https://github.com/NationalSecurityAgency/ghidra/releases (дата звернення: 07.07.2023).

References:

1. Starenkyi, I., & Donchenko, O. (2022). Doslidzhennia prohramnoho zabezpechennia “Simple Games” ta “Iconnect” dlia operatsiinoi systemy “Linux” [Study of software “Simple Games” and “Iconnect” for operating system “Linux”]. *Visnyk ONDISE*. Vypusk 12. P. 74–93 [in Ukrainian].

2. Wikipedia (2023). Visual Studio. *Wikipedia: the free encyclopedia*. Retrieved from: https://en.wikipedia.org/wiki/Visual_Studio [in English].

3. N. a. (n. d.). Microsoft Network Monitor 3.4 (archive). Microsoft: official website. Retrieved from: https://www.microsoft.com/en-us/download/4865 [in English].

4. National Security Agency (n. d.). Ghidra. Retrieved from: https://github.com/NationalSecurityAgency/ghidra/releases [in English].