

# Panduan Riset Komprehensif untuk Pengembangan Pemindai Dungeon Albion Online Generasi Berikutnya

Laporan ini menyajikan panduan riset langkah demi langkah yang mendetail, dirancang untuk mengumpulkan semua informasi yang diperlukan guna membangun pemindai (scanner) dungeon Albion Online yang canggih. Fokus utama adalah pada deteksi Peti (Chest) dan Mob yang akurat di dalam dungeon Solo, Group, dan khususnya Avalonian. Panduan ini bertujuan untuk mengatasi masalah yang dihadapi dalam proyek pemindai sebelumnya, seperti paket jaringan yang tidak ditemukan, akurasi yang tidak tepat, kesalahan pemindaian, error berkepanjangan, dan kegagalan deteksi, dengan memanfaatkan referensi dan data dari berbagai repositori komunitas yang telah teridentifikasi.

## I. Pengetahuan Dasar dan Pengaturan

Sebelum memulai pengembangan pemindai dungeon Albion Online, penting untuk membangun fondasi pengetahuan yang kuat dan menyiapkan lingkungan pengembangan yang sesuai. Tahap ini krusial untuk memastikan kelancaran proses riset dan pengembangan selanjutnya.

### A. Lingkungan Pengembangan dan Bahasa Pemrograman Esensial

Pemilihan bahasa pemrograman dan Lingkungan Pengembangan Terpadu (IDE) yang tepat merupakan langkah awal yang fundamental. Berdasarkan analisis terhadap proyek-proyek referensi, dua bahasa pemrograman utama muncul sebagai kandidat kuat: **Python** dan **C#**. Python digunakan dalam proyek seperti imjangkar/albibong<sup>1</sup> dan CleverCheat/albion-dungeon-scanner<sup>2</sup> (teridentifikasi melalui penggunaan perintah `pdm install` dan `pdm run scan`). Sementara itu, C# adalah bahasa asli dari Photon Engine<sup>3</sup>, yang digunakan oleh Albion Online, dan juga merupakan bahasa yang dipakai dalam berbagai alat bantu komunitas seperti pypye/albion-dungeon-checker<sup>4</sup> dan JPCodeCraft/AlbionDataAvalonia.<sup>7</sup>

Pilihan bahasa akan memengaruhi ketersediaan pustaka untuk *packet sniffing* (pengendus paket), penguraian protokol Photon, dan pengembangan antarmuka pengguna. Python menawarkan pustaka yang matang seperti scapy untuk manipulasi paket tingkat lanjut dan akses ke versi photon-packet-parser berbasis Python.<sup>1</sup> Di sisi lain, C# memiliki pustaka seperti SharpPcap<sup>10</sup> dan memungkinkan interaksi yang lebih langsung atau bahkan *native* dengan struktur protokol Photon. Mengingat proyek sebelumnya pengguna (athernot/albion-dungeon-scanner, kemungkinan berbasis Python berdasarkan deskripsi masalah dan referensi ke CleverCheat/albion-dungeon-scanner<sup>2</sup>) mengalami kendala seperti "paket network

tidak ditemukan" dan "akurasi tidak tepat", evaluasi cermat terhadap bahasa pemrograman menjadi penting. Meskipun Python terbukti mampu menangani tugas ini (seperti yang ditunjukkan oleh imjangkar/albibong <sup>1</sup>), C# mungkin menawarkan solusi yang lebih stabil atau berkinerja lebih tinggi untuk penanganan paket Photon karena kedekatannya dengan ekosistem Photon.<sup>3</sup> Keputusan ini harus mempertimbangkan kurva pembelajaran, ketersediaan pustaka yang relevan, dan potensi untuk mengatasi kegagalan proyek sebelumnya.

Penggunaan sistem kontrol versi seperti Git sangat krusial, dan pengguna diasumsikan sudah familiar dengan GitHub mengingat banyaknya referensi ke repositori di platform tersebut.

## B. Alat Bantu Esensial untuk Analisis Jaringan

Beberapa alat bantu esensial akan sangat diperlukan selama proses riset dan pengembangan:

1. **Wireshark:** Alat ini tidak dapat dinegosiasikan. Wireshark akan menjadi instrumen utama untuk menangkap, melihat, dan menganalisis secara manual paket jaringan mentah. Pemahaman mendalam tentang struktur paket sebelum mencoba penguraian otomatis sangat bergantung pada alat ini. Kegunaannya sebagai alat generik untuk *sniffing* telah diakui dalam diskusi komunitas.<sup>12</sup>
2. **Npcap (Windows) / libpcap (Linux/macOS):** Ini adalah pustaka penangkap paket dasar yang menjadi dependensi bagi sebagian besar alat *sniffing* dan pustaka pemrograman seperti SharpPcap dan Scapy. Proyek imjangkar/albibong <sup>1</sup> menggunakan Scapy, yang pada gilirannya memerlukan Npcap/libpcap. Demikian pula, Triky313/AlbionOnline-StatisticsAnalysis secara eksplisit menyebutkan instalasi Npcap.<sup>13</sup> Kegagalan dalam instalasi atau konfigurasi Npcap/libpcap yang benar, interferensi dari *firewall*, atau ketidakmampuan pustaka *sniffing* untuk mengidentifikasi antarmuka jaringan yang benar atau memfilter lalu lintas Albion Online bisa menjadi akar masalah "paket network tidak ditemukan" yang dialami sebelumnya.
3. **Editor Teks / Fitur IDE:** Alat dengan kemampuan penampil/editor heksadesimal yang baik dan dukungan untuk bahasa pemrograman yang dipilih akan penting untuk memeriksa data paket dan mengembangkan skrip penguraian.

Kemampuan untuk memfilter lalu lintas secara efektif di Wireshark (misalnya, berdasarkan port UDP 5056, seperti yang diindikasikan oleh contoh dari DocTi/albion-network <sup>15</sup> dan OxNOx/photon-packet-parser <sup>16</sup>) adalah prasyarat untuk mengurangi *noise* dan fokus pada paket game yang relevan. Keterampilan ini akan

langsung diterapkan saat mengonfigurasi filter secara terprogram.

### C. Gambaran Umum Arsitektur Jaringan Albion Online

Pemahaman dasar tentang arsitektur jaringan Albion Online sangat penting:

1. **Model Client-Server:** Semua status game yang krusial (posisi pemain, status mob/peti) dikelola secara otoritatif oleh server. Klien mengirimkan permintaan (operasi) dan menerima pembaruan (event).<sup>12</sup>
2. **Photon Engine:** Albion Online menggunakan Photon Engine untuk komunikasi *multiplayer real-time*.<sup>3</sup> Ini berarti lalu lintas akan mengikuti protokol Photon, khususnya Protocol16.<sup>8</sup>
3. **Protokol UDP:** Photon utamanya menggunakan UDP untuk komunikasi berlatensi rendah.<sup>3</sup> Lalu lintas game umumnya berjalan pada port UDP 5056.<sup>15</sup>
4. **Sebagian Besar Plain-Text:** Meskipun Photon memiliki serialisasi biner sendiri, data spesifik game dalam paket Photon sebagian besar tidak terenkripsi atau terobfusikasi, seperti yang dinyatakan dalam diskusi komunitas.<sup>12</sup> Inilah yang memungkinkan *sniffing* menjadi metode yang layak.

Model "server adalah otoritatif" berarti pemindai pada dasarnya bergantung pada apa yang *diberitahukan* server kepada klien. Perbedaan atau penundaan dalam pembaruan server dapat menyebabkan ketidakakuratan yang dirasakan oleh pemindai. Fakta bahwa data bersifat "plain-text" dalam struktur Photon<sup>12</sup> adalah landasan fundamental dari seluruh upaya ini. Jika data dienkripsi secara berat dengan kunci per sesi, tingkat kesulitan proyek akan meningkat secara eksponensial. Hal ini menyederhanakan proses rekayasa balik secara signifikan; tantangan utamanya adalah memahami format serialisasi Photon (Protocol16) dan arti dari kode operasi/event spesifik beserta parameternya, bukan memecahkan proteksi kriptografi.

## II. Menguasai Intersepsi & Analisis Lalu Lintas Jaringan

Setelah fondasi dan pemahaman dasar terbentuk, langkah selanjutnya adalah menguasai teknik untuk mencegat dan menganalisis lalu lintas jaringan game Albion Online secara efektif.

### A. Teknik untuk Mengendus Paket Albion Online

Bagian ini akan merinci metode untuk menangkap lalu lintas game secara andal. Hal ini melibatkan:

1. **Identifikasi Antarmuka Jaringan yang Benar:** Ini sangat penting, terutama dengan adanya VPN atau beberapa adapter jaringan. Triky313/AlbionOnline-StatisticsAnalysis menyebutkan Npcap dengan pemfilteran

- untuk pengguna VPN dan kemampuan memilih perangkat jaringan tertentu.<sup>18</sup>
2. **Pemfilteran Lalu Lintas:** Fokus pada port UDP 5056, yang secara konsisten digunakan oleh lalu lintas Photon Albion Online.<sup>15</sup>
  3. **Hak Akses yang Diperlukan:** Menjalankan alat/skrip penangkap paket dengan izin yang diperlukan (misalnya, administrator/root), seperti yang dicatat untuk Scapy di macOS dalam dokumentasi imjangkar/albibong.<sup>1</sup>

Potensi masalah yang mungkin timbul meliputi *firewall* yang memblokir penangkapan paket, VPN yang merutekan lalu lintas melalui antarmuka virtual yang mungkin tidak dipantau secara *default*, dan aplikasi lain yang menghasilkan lalu lintas UDP yang dapat membingungkan jika pemfilteran tidak tepat. Masalah "paket network tidak ditemukan" dari proyek pengguna sebelumnya secara langsung menunjuk pada kegagalan dalam fase penangkapan awal ini. Solusi yang tangguh perlu mengatasi pemilihan antarmuka dan potensi batasan OS/firewall. Meskipun port 5056 tampaknya standar untuk Photon, pembaruan game secara teoretis dapat mengubahnya. Idealnya, pemindai harus membuat port ini dapat dikonfigurasi atau memiliki cara untuk memverifikasinya.

## B. Memilih dan Mengimplementasikan Pustaka Penangkap Paket

Pilihan pustaka penangkap paket akan sangat bergantung pada bahasa pemrograman utama yang dipilih pada Tahap I.A.

### Untuk Python:

- **Scapy:** Sangat kuat untuk manipulasi dan pembuatan paket, tetapi bisa memiliki kurva belajar yang lebih curam untuk tugas *sniffing* sederhana. Digunakan oleh imjangkar/albibong.<sup>1</sup> Membutuhkan Npcap/libpcap.
- **Pcap/Dpkt (atau sejenisnya):** Alternatif yang lebih ringan untuk penangkapan dan penguraian murni jika Scapy dianggap berlebihan.
- **python-pcapng atau python-pyshark (pembungkus Wireshark tshark):** Dapat digunakan untuk kapabilitas yang lebih mirip Wireshark secara langsung.

### Untuk C#:

- **SharpPcap:** Pembungkus.NET yang populer untuk libpcap/WinPcap, menyediakan akses ke data paket mentah.<sup>10</sup> Digunakan dalam DocTi/albion-network<sup>15</sup> dan pypye/albion-dungeon-checker.<sup>5</sup>
- **Pcap.Net:** Pustaka.NET komprehensif lainnya untuk WinPcap, menawarkan diseksi paket yang mendetail.

Proyek pengguna sebelumnya (athernot/albion-dungeon-scanner) kemungkinan

menggunakan pustaka Python. Jika masalah muncul dari pustaka itu sendiri (misalnya, kinerja, bug, kompleksitas), beralih atau meningkatkan versi mungkin bermanfaat. imjangkar/albibong<sup>1</sup> berhasil menggunakan Scapy, yang menunjukkan kelayakan Python. Pustaka C# seperti SharpPcap sudah mapan untuk alat Albion Online.<sup>5</sup> Penggunaan langsung mereka dalam proyek lain yang berhasil atau dapat dirujuk (pypye/albion-dungeon-checker, DocTi/albion-network) menjadikannya pertimbangan kuat jika C# dipilih.

Berikut adalah tabel perbandingan beberapa pustaka penangkap paket yang relevan:

Nama Pustaka	Bahasa	Fitur Utama	Dependensi	Contoh Proyek Albion yang Menggunakan
Scapy	Python	Manipulasi paket tingkat lanjut, <i>sniffing</i> , diseksi, pembuatan paket	Npcap/libpcap	imjangkar/albibong <sup>1</sup>
SharpPcap	C#	Pembungkus.NET untuk libpcap/WinPcap, akses data paket mentah, filter	Npcap/WinPcap	pypye/albion-dungeon-checker <sup>5</sup> , DocTi/albion-network <sup>15</sup>
Pcap.Net	C#	Pembungkus.NET untuk WinPcap, diseksi paket detail, pembuatan paket	WinPcap	(Kurang umum dalam referensi yang diberikan)
Pcapy / Dpkt	Python	Pcapy untuk penangkapan, Dpkt untuk penguraian, lebih ringan dari Scapy	Npcap/libpcap	(Alternatif umum untuk Python)

Tabel ini bertujuan untuk membantu dalam membuat keputusan yang terinformasi

mengenai pustaka mana yang akan digunakan, dengan mempertimbangkan trade-off antara kemudahan penggunaan, fitur, dan rekam jejak dalam komunitas Albion Online.

### C. Pemfilteran Paket Awal dan Identifikasi Sesi

Selain port UDP 5056, pemfilteran lebih lanjut mungkin diperlukan (misalnya, berdasarkan alamat IP server Albion Online, meskipun ini dapat berubah). Mengidentifikasi "sesi" atau koneksi game yang berbeda jika beberapa klien dijalankan atau jika terjadi koneksi ulang juga penting. Paket Photon seringkali berisi PeerID<sup>19</sup>, yang dapat membantu dalam manajemen sesi. Tujuannya adalah untuk mengisolasi aliran paket yang relevan dengan instans game aktif yang sedang dipindai.

Alamat IP dinamis untuk server game berarti pemfilteran berbasis IP perlu adaptif, mungkin dengan pencarian DNS awal atau dengan mengamati pola lalu lintas. Penanganan pemutusan dan koneksi ulang secara anggun penting untuk pemindai yang mungkin berjalan untuk periode yang lama. Masalah "error berkepanjangan" pada proyek sebelumnya bisa jadi terkait dengan pemutusan koneksi yang tidak ditangani. Interupsi jaringan adalah hal yang umum. Pemindai yang tangguh harus mendeteksi ini, berpotensi membersihkan statusnya saat ini untuk sesi tertentu, dan melakukan inisialisasi ulang ketika lalu lintas dilanjutkan. Event OnStatusChanged dari Photon<sup>20</sup> mengisyaratkan bagaimana klien menangani ini, dan *sniffer* mungkin dapat menyimpulkan perubahan status serupa.

## III. Dekonstruksi Protokol Photon (Protocol16)

Memahami cara kerja Photon Engine dan protokol komunikasinya adalah inti dari pengembangan pemindai yang akurat. Albion Online menggunakan Protocol16 dari Photon.

### A. Memahami Dasar-dasar Photon Engine: Operasi, Event, dan Parameter

- **Gambaran Umum Photon:** Photon adalah kerangka kerja untuk game *multiplayer real-time*.<sup>3</sup>
- **Operasi (Permintaan):** Tindakan yang diinisiasi klien dan dikirim ke server (misalnya, Pindah, GunakanMantra, GabungRuangan). Diidentifikasi oleh OperationCode (byte).<sup>3</sup> Parameter dikirim dalam struktur mirip kamus (dictionary).<sup>3</sup>
- **Event:** Pesan yang diinisiasi server dan dikirim ke klien (misalnya, PemainLainBergerak, MobMuncul, EfekMantra). Diidentifikasi oleh EventCode (byte).<sup>3</sup> Parameter juga dalam struktur kamus.<sup>3</sup>
- **Parameter:** Pasangan kunci-nilai yang membawa data untuk operasi dan event.

Kunci biasanya berupa byte, nilai dapat berupa berbagai tipe yang didukung Photon.<sup>3</sup> Kunci parameter 253 (untuk opcode OperationResponse) dan 252 (untuk opcode EventData, jika tidak ada di event.Code) dicatat dalam.<sup>20</sup>

- **Protocol16:** Versi spesifik dari protokol biner Photon yang digunakan. Ini mendefinisikan bagaimana tipe data diserialisasi/deserialisasi.<sup>8</sup> Enum GpType mendefinisikan tipe-tipe ini.<sup>17</sup>

Pemahaman yang jelas tentang perbedaan antara Operasi (klien->server) dan Event (server->klien) adalah fundamental. Pemindai akan terutama tertarik untuk mengurai **Event** dari server, karena ini mengumumkan perubahan di dunia game (kemunculan mob/peti, pergerakan pemain). Struktur kamus untuk parameter<sup>3</sup> berarti penguraian melibatkan pengetahuan tentang kunci byte untuk bidang data spesifik dalam setiap event/operasi. Ini adalah bagian utama dari upaya rekayasa balik.

## B. Mengidentifikasi Kode Operasi (OpCodes) dan Kode Event Inti Albion Online

Ini adalah langkah riset yang krusial. Tujuannya adalah untuk memetakan kode numerik ke tindakan game.

Sumber utama meliputi:

- kolloko2/AlbionOnlinePhotonEventIds.git (Referensi pengguna)
- W4RPWISH/Albion\_Op-EvCodes\_Checker.git<sup>94</sup>
- Oblu/AlbionTracker/blob/master/Albion.Common/Photon/OperationCodes.cs<sup>21</sup> dan EventCodes.cs.<sup>26</sup> Ini kemungkinan adalah definisi enum C#.
- Triky313/AlbionOnline-StatisticsAnalysis<sup>14</sup> mengonfirmasi relevansinya].
- Jadd/Afterglow.Albion.Packet.cs<sup>20</sup> menunjukkan contoh penanganan opcode spesifik dan ekstraksi opcode dari kamus parameter.

Metodologi yang disarankan:

1. Kumpulkan semua daftar kode yang tersedia dari repositori ini.
2. Referensi silang dan gabungkan, catat perbedaan atau perbedaan versi.
3. Gunakan Wireshark bersamaan dengan gameplay: lakukan tindakan spesifik (masuk dungeon, bunuh mob, buka peti) dan amati paket yang sesuai untuk memvalidasi kode yang diketahui dan menemukan yang baru.
4. Fokus pada kode yang relevan dengan: Pergerakan pemain, masuk/keluar dungeon, kemunculan/penghilangan/kematian mob, kemunculan/pembukaan/penghilangan peti, posisi entitas.

Sifat daftar kode yang didorong oleh komunitas ini berarti mereka dapat menjadi usang dengan pembaruan game. Masalah "Not Working after Foundations Update" untuk CleverCheat/albion-dungeon-scanner<sup>29</sup> adalah contoh utama. Verifikasi



berkelanjutan diperlukan. Beberapa kode adalah kode Photon generik (misalnya, Ping, Join <sup>21</sup>), sementara yang lain khusus untuk Albion Online. Pemindai perlu membedakan dan fokus pada yang spesifik game untuk entitas dungeon.

Berikut adalah tabel yang mengkategorikan jenis-jenis kode Photon yang relevan:

Kategori	Contoh OpCode/EventCode Albion (Numerik)	Kemungkinan Parameter (Konseptual)	Sumber Repositori/Observasi
Aksi Pemain (Gerakan, Mantra)	Move (OpCode 18 <sup>21</sup> ), CastStart (OpCode 20 <sup>21</sup> )	ID Pemain, Koordinat (X,Y,Z), ID Mantra, ID Target	Oblu/AlbionTracker <sup>21</sup> , Observasi
Siklus Hidup Mob (Muncul, Mati)	Event NewCharacter (parameter tipe mob), Event Kematian Mob	ID Mob, Tipe Mob (UniqueName), Posisi, Status Kesehatan	Jadd/Afterglow <sup>20</sup> , Observasi
Siklus Hidup Peti (Muncul, Dibuka)	Event Kemunculan Peti, Event Peti Dibuka	ID Peti, Tipe Peti (UniqueName), Posisi, Status	Observasi, pypye/albion-dungeon-checker <sup>6</sup>
Transisi Zona/Dungeon (Masuk, Ganti Lantai)	ChangeCluster (OpCode 28 <sup>21</sup> )	ID Pemain, ID Cluster Baru, Mungkin ID Lantai	Oblu/AlbionTracker <sup>21</sup> , Observasi
Pembaruan Posisi Entitas	Event Pembaruan Posisi	ID Entitas, Koordinat Baru (X,Y,Z)	Observasi
Interaksi Inventaris	InventoryMoveItem (OpCode 26 <sup>21</sup> )	ID Item, Slot Sumber, Slot Tujuan	Oblu/AlbionTracker <sup>21</sup>

Tabel ini memberikan gambaran terstruktur tentang jenis kode yang perlu ditangani oleh pemindai, menghubungkan nomor kode abstrak dengan event game konkret dan jenis data yang kemungkinan dibawanya. Ini membantu memprioritaskan upaya rekayasa balik pada kode yang paling relevan dengan pemindaian dungeon.

### C. Deserialisasi Tipe Data Photon

Photon Protocol16 menggunakan kode tipe spesifik (GpType) untuk melakukan



serialisasi data.<sup>17</sup> Tipe umum meliputi byte, bool, short, int, long, float, double, string, byte, object, Hashtable, Dictionary<object,object> (dan variannya).<sup>17</sup> Struktur data seperti Hashtable/Dictionary dan Array sangat penting untuk membawa parameter dalam Operasi dan Event. Proses deserialisasi melibatkan pembacaan kode tipe, kemudian membaca data sesuai dengan format dan panjang tipe tersebut. Metode Protocol16.Deserialize() adalah pusatnya.<sup>17</sup>

Deserialisasi yang salah adalah sumber utama error dan "akurasi tidak tepat." Memahami urutan byte yang tepat (*endianness*, biasanya Big Endian untuk jaringan<sup>30</sup>) dan panjang tipe sangat penting. Jika int (4 byte) dibaca sebagai short (2 byte), atau jika *endianness* yang salah diasumsikan, semua data berikutnya dalam paket akan salah dibaca, yang menyebabkan error berantai dan data yang salah. Ini kemungkinan merupakan kontributor masalah proyek sebelumnya. Struktur bersarang (misalnya, array kamus, di mana setiap kamus mewakili mob dengan propertinya) umum terjadi dan memerlukan deserialisasi rekursif yang cermat.

#### **D. Memanfaatkan Pustaka photon-packet-parser**

Pengguna menyebutkan imjangkar/albibong<sup>1</sup>, yang menggunakan photon-packet-parser yang dimodifikasi. Ada versi JavaScript (OxNOx/photon-packet-parser<sup>16</sup>) dan Python (santiac89/photon-packet-parser<sup>8</sup>) yang tersedia, keduanya terinspirasi atau berdasarkan implementasi C#.<sup>8</sup> Pustaka ini bertujuan untuk menyederhanakan deserialisasi paket Protocol16.

Tugas riset adalah memeriksa kode sumber parser ini (terutama yang digunakan oleh albibong atau sumber aslinya) untuk memahami bagaimana mereka menangani GpType yang berbeda, bagaimana mereka memetakan kunci byte dalam kamus parameter ke nama yang bermakna (jika ada), dan kelengkapan serta akurasi untuk penggunaan Photon spesifik Albion. santiac89/photon-packet-parser<sup>8</sup> secara eksplisit menyatakan bahwa ia tidak lengkap dan "berdasarkan pustaka C# yang dikodekan oleh Oblu," yang menunjukkan bahwa karya Oblu (seperti AlbionTracker) mungkin merupakan sumber yang lebih fundamental untuk memahami logika deserialisasi C#.

Parser ini dapat menjadi awal yang signifikan tetapi mungkin tidak sempurna atau sepenuhnya mutakhir, seperti yang diperingatkan oleh README santiac89/photon-packet-parser.<sup>8</sup> "Modifikasi" yang disebutkan oleh albibong<sup>1</sup> bisa jadi merupakan perbaikan penting atau adaptasi spesifik Albion. Memahami logika inti parser ini, bahkan jika tidak menggunakannya secara langsung, memberikan wawasan berharga tentang bagaimana deserialisasi Protocol16 diimplementasikan secara

praktis.

## IV. Mengekstrak dan Memanfaatkan File Data Game

Selain data jaringan *real-time*, pemindai memerlukan akses ke data game statis untuk memberikan konteks pada entitas yang terdeteksi.

### A. Mengakses Data Aset Game

- **Selivaniuk/albionDataServer.git**<sup>33</sup>: Repositori ini, yang dirujuk pengguna untuk basis data, kemungkinan merupakan proyek yang memproses dan menyajikan data game, mungkin berasal dari ao-bin-dumps. Sifat pastinya perlu diselidiki jika dapat diakses, atau disimpulkan dari proyek yang mungkin menggunakannya.
- **ao-bin-dumps** (organisasi GitHub: ao-data, repositori spesifik ao-data/ao-bin-dumps atau broderickhyman/ao-bin-dumps<sup>34</sup>): Ini adalah proyek komunitas terkenal yang menyediakan *dump* file game biner Albion Online (misalnya, items.bin, world.bin) yang dikonversi ke format yang lebih dapat digunakan seperti JSON dan XML. File-file ini berisi data statis tentang item, mob, lokasi, dll..<sup>13</sup> File .bin mentah adalah bagian dari instalasi klien game Albion Online.

ao-bin-dumps adalah sumber de facto untuk data game statis bagi banyak alat pihak ketiga. Keandalan dan frekuensi pembaruannya adalah kunci. Jika ao-bin-dumps tidak diperbarui segera setelah patch game besar, alat yang mengandalkannya untuk definisi item/mob mungkin menunjukkan informasi yang usang atau salah.

Selivaniuk/albionDataServer mungkin menawarkan basis data yang sudah diproses sebelumnya dan terstruktur yang berasal dari *dump* ini, yang berpotensi menyederhanakan akses data jika skemanya sesuai dan selalu diperbarui.

### B. Alat Bantu untuk Mengurai File .bin

- **albiondata-bin-dumper** (dari ao-data atau broderickhyman<sup>40</sup>): Alat ini mengonversi file .bin milik game menjadi XML dan JSON. Kemungkinan ini adalah sumber file yang ditemukan di ao-bin-dumps. Memahami penggunaannya dapat membantu jika pemrosesan langsung file .bin diperlukan atau untuk memahami asal *dump* JSON/XML.
- **ao-id-extractor** (dari ao-data atau broderickhyman<sup>41</sup>): Secara khusus mengekstrak ID dan nama dari Items.bin dan world.bin. Ini sudah usang dan digantikan oleh albiondata-bin-dumper.<sup>41</sup> Beberapa alat mungkin menggunakan teknik ekstraksi aset Unity<sup>38</sup> jika penguraian .bin langsung terlalu rumit, meskipun albiondata-bin-dumper tampaknya menjadi standar.

Meskipun pemindai kemungkinan akan mengonsumsi file JSON/XML yang sudah

dikonversi dari ao-bin-dumps, mengetahui tentang albiondata-bin-dumper berguna untuk memahami asal data dan untuk potensi kebutuhan di masa depan jika penguraian kustom file .bin menjadi perlu (misalnya, jika pembaruan ao-bin-dumps tertunda). Format output (JSON/XML) dari albiondata-bin-dumper menentukan bagaimana pemindai perlu mengurai data statis ini.

### C. Memahami Struktur Data (dari ao-bin-dumps JSON/XML)

Ini melibatkan pemeriksaan detail struktur dan bidang dalam file JSON/XML kunci. Subdirektori formatted/ di ao-bin-dumps seringkali lebih disukai.

- **items.json**<sup>13</sup>:
  - Bidang kunci: UniqueName (ID internal), LocalizedNames (untuk berbagai bahasa), ItemValue, Enchantments, SlotType (untuk peralatan), MaxStackSize, Tier, Weight, ShopCategory, ShopSubCategory1.
  - Untuk peralatan: daftar mantra/kemampuan (ενεργησκιρ), statistik (CraftingRequirements, FoodValue untuk barang habis pakai).
  - Untuk peti: UniqueName akan mengidentifikasi tipe peti (misalnya, TREASURE\_CHEST\_SOLO\_STANDARD, TREASURE\_CHEST\_AVALONIAN\_RARE). Informasi tabel *loot* spesifik umumnya *tidak* ada di items.json tetapi ditentukan oleh logika game dan berpotensi file lain atau konfigurasi sisi server. Namun, item peti itu sendiri akan didefinisikan di sini.<sup>6</sup> dan<sup>47</sup> mencantumkan kelangkaan peti (Hijau, Biru, Ungu, Emas/Legendaris, Putih untuk Avalonian).
- **mobs.json**<sup>48</sup>:
  - Bidang kunci: UniqueName, LocalizedNames, Faction, Spells/Abilities (referensi ke ID mantra), Health, Damage, AttackType, MovementSpeed, RespawnTimeMin/Max (<sup>54</sup> menyiratkan data ini ada di suatu tempat).
  - Info terkait *loot*: Mungkin berisi referensi ke tabel *loot* atau kategori, tetapi peluang *drop* yang tepat biasanya ada di sisi server. Tipe mob yang relevan dengan dungeon akan dapat diidentifikasi berdasarkan UniqueName atau faksi mereka (misalnya, "AVALONIAN\_ACOLYTE"<sup>55</sup>).
- **world.json / cluster.json**<sup>45</sup>:
  - world.json seringkali berisi array klaster (zona peta). Setiap klaster memiliki ID, Name, Type (AMAN, KUNING, MERAH, BLACKZONE\_ROYAL, BLACKZONE\_OUTLANDS).
  - Mungkin berisi koordinat untuk pintu masuk dungeon statis, koneksi ke klaster lain, distribusi sumber daya, informasi wilayah.<sup>58</sup> menjelaskan tipe zona dan wilayah.<sup>65</sup> menjelaskan properti portal/zona Roads of Avalon, yang mungkin tercermin dalam world.json atau file Avalon khusus.
  - Informasi tentang lokasi spesifik *di dalam* dungeon dinamis/acak cenderung

tidak ada di sini dan lebih mungkin ada di `randomizeddungeons.json` atau berasal dari paket jaringan.

- **`randomizeddungeons.json`** <sup>66</sup>, **`corrupteddungeons.json`** <sup>70</sup>:
  - Ini sangat penting untuk memahami struktur dungeon.
  - `randomizeddungeons.json`: Kemungkinan berisi definisi "ubin" atau "modul" dungeon, titik koneksi di antaranya, potensi lokasi kemunculan mob/peti di dalam ubin, parameter penskalaan kesulitan, konfigurasi ruang bos.
  - `corrupteddungeons.json`: Spesifik untuk dungeon terkorupsi seperti tipe/lokasi kuil, tipe mob unik untuk invasi (misalnya, Corrupted Mephitis <sup>72</sup>), mekanisme invasi.
  - <sup>72</sup> dan <sup>72</sup> menjelaskan mekanisme dan mob Corrupted Dungeon. <sup>55</sup> menjelaskan Avalonian Dungeons. File JSON harus mencerminkan elemen game ini.

Bidang `UniqueName` sangat penting untuk melakukan referensi silang entitas yang diidentifikasi dalam paket jaringan (yang biasanya membawa semacam ID) dengan data statisnya (nama, statistik, tipe) dari file JSON ini. Paket jaringan kemungkinan akan menyediakan ID untuk mob atau peti. ID ini perlu dicari di `items.json` atau `mobs.json` (menggunakan `UniqueName` sebagai kunci) untuk mendapatkan nama yang dapat dibaca manusia, tingkatan, kelangkaan, kemampuan, dll. Kaitan ini penting agar pemindai dapat memberikan informasi yang berguna. Tabel *loot* bersifat kompleks dan kemungkinan tidak dirinci sepenuhnya dalam file sisi klien. Pemindai dapat mengidentifikasi *tipe* dan *kelangkaan* peti, tetapi memprediksi *loot* yang tepat umumnya di luar kemampuan *sniffing* pasif dan seringkali ditentukan server.<sup>47</sup> Informasi tata letak dungeon di `randomizeddungeons.json` mungkin bersifat tingkat tinggi (definisi ubin, aturan koneksi). Tata letak aktual yang diinstansiasi dengan posisi mob/peti spesifik untuk instans dungeon *langsung* akan terutama berasal dari event jaringan. `randomizeddungeons.json` menyediakan "cetak biru" atau "palet" untuk dungeon. Event jaringan menyediakan "lukisan langsung" dari instans dungeon tertentu. Pemindai membutuhkan keduanya: cetak biru untuk memahami apa yang *dapat* muncul, dan event jaringan untuk melihat apa yang *telah* muncul.

Berikut adalah ringkasan file data game utama dan relevansinya:

Nama File	Poin Data Kunci yang Relevan untuk Pemindai	Cara Membantu Pemindaian
<code>items.json</code>	<code>UniqueName</code> , <code>LocalizedNames</code> , tipe item (peti, peralatan), kelangkaan	Menerjemahkan ID peti dari paket menjadi nama, kelangkaan, dan detail item

	peti	lainnya.
mobs.json	UniqueName, LocalizedNames, faksi, statistik (HP, damage), kemampuan	Menerjemahkan ID mob dari paket menjadi nama mob, statistik, dan kemampuan.
world.json/cluster.json	ID klaster, nama, tipe zona, mungkin pintu masuk dungeon statis	Memberikan konteks tentang zona tempat dungeon berada.
randomizeddungeons.json	Definisi ubin dungeon, aturan koneksi, potensi lokasi spawn	Membantu memahami struktur potensial dungeon acak.
corrupteddungeons.json	Mekanisme invasi, tipe mob/kuil spesifik	Menyediakan data khusus untuk memindai Corrupted Dungeons secara akurat.
localization (TMX/JSON)	Pemetaan UniqueName ke teks yang dilokalkan	Menampilkan nama item, mob, dll., yang dapat dibaca manusia dalam bahasa yang dipilih.

Tabel ini memberikan referensi cepat, menghubungkan file dengan tujuannya dalam konteks membangun pemindai dan membantu mengatur strategi ekstraksi data.

#### D. Mengintegrasikan Data Lokalisasi

- **JPCoDeCraft/AlbionLocalization.git**<sup>7</sup>: Repositori ini dimaksudkan untuk menyediakan pemetaan dari ID game internal/UniqueName ke nama yang dapat dibaca manusia dalam berbagai bahasa.
- **Format TMX**: Albion Online kemungkinan menggunakan TMX (Translation Memory eXchange), standar berbasis XML untuk data lokalisasi.<sup>75</sup>
  - Struktur: Root <tmx>, <header> dengan metadata, <body> berisi elemen <tu> (unit terjemahan). Setiap <tu> memiliki tuid (ID unit terjemahan, kemungkinan UniqueName atau ID internal) dan beberapa elemen <tuv> (varian unit terjemahan), satu untuk setiap bahasa (misalnya, <tuv xml:lang="en"> <seg>Nama Inggris</seg> </tuv>).<sup>77</sup>
- Pemindai akan menggunakan ini untuk menampilkan nama yang tepat untuk item, mob, dll., alih-alih ID internal.

Bidang LocalizedNames yang sering ditemukan dalam file JSON ao-bin-dumps mungkin sudah berisi data lokalisasi yang sudah diproses sebelumnya, yang berpotensi mengurangi kebutuhan langsung untuk mengurai file TMX yang kompleks jika bahasa target tercakup. Jika items.json sudah memiliki objek LocalizedNames dengan kunci seperti "EN-US", "DE-DE", dll., untuk setiap item, maka JPCodeCraft/AlbionLocalization mungkin menjadi redundan atau berfungsi sebagai sumber yang lebih komprehensif/mutakhir jika ao-bin-dumps tertinggal dalam pembaruan lokalisasi. Hal ini perlu diverifikasi. Mempertahankan sinkronisasi antara ID internal yang digunakan dalam paket jaringan/file data game dan tuid dalam file lokalisasi sangat penting untuk tampilan yang benar. Jika pemindai mengidentifikasi mob dengan MOB\_ID\_XYZ dari sebuah paket, ia perlu menemukan ID yang sama persis ini dalam file TMX (atau bidang LocalizedNames) untuk mengambil nama tampilannya. Ketidakcocokan apa pun akan menghasilkan nama yang hilang atau salah.

Jika penguraian file TMX secara langsung diperlukan, pemahaman tentang strukturnya sangat penting. Tabel berikut, berdasarkan <sup>77</sup>, memberikan contoh yang jelas tentang bagaimana string game disimpan dan diterjemahkan, memungkinkan pengembangan parser TMX.

Elemen TMX	Contoh Atribut	Tujuan dalam Lokalisasi Albion
<tmx>	version="1.4"	Elemen root untuk file TMX, menentukan versi format.
<header>	creationtool, srclang="en", adminlang="en"	Berisi metadata tentang file TMX, seperti alat pembuatan dan bahasa sumber.
<body>		Kontainer untuk semua unit terjemahan.
<tu>	tuid="UNIQUE_ITEM_NAME_001"	Unit Terjemahan: Mewakili satu segmen teks sumber dan terjemahannya. tuid adalah pengidentifikasi unik.
<tuv>	xml:lang="en" atau xml:lang="es"	Varian Unit Terjemahan: Berisi segmen teks untuk bahasa

		tertentu.
<seg>		Segmen: Berisi teks aktual yang diterjemahkan atau sumber.

## V. Mengidentifikasi dan Melacak Entitas Dungeon

Setelah data jaringan dan data game statis dapat diakses dan dipahami, langkah berikutnya adalah menggabungkannya untuk mengidentifikasi dan melacak entitas di dalam dungeon secara akurat.

### A. Mengorelasikan Event Jaringan dengan Tindakan Dalam Game

Di sinilah data jaringan bertemu dengan logika game. Tugas inti adalah mengidentifikasi Event Photon spesifik yang menandakan:

- Pemain masuk/keluar dungeon atau berganti lantai (misalnya, OpCode ChangeCluster<sup>21</sup> atau event kustom).
- Entitas (mob, peti) muncul, menghilang, atau berubah status.

Ini memerlukan observasi yang cermat di Wireshark selama gameplay, memicu tindakan ini dan mencatat paketnya. athernot/albion-dungeon-scanner memiliki masalah "miss scanning" dan "tidak bisa mendeteksi", yang menunjukkan masalah dalam langkah korelasi ini. CleverCheat/albion-dungeon-scanner juga memiliki masalah deteksi.<sup>29</sup>

ID Entitas adalah fundamental. Sebagian besar event terkait entitas akan membawa ID. ID ini adalah kunci untuk melacak siklus hidup entitas dan menghubungkannya dengan data statis dari file game. Ketika mob muncul, sebuah event kemungkinan akan berisi ID instans uniknya dan ID tipenya (UniqueName). ID instans digunakan untuk melacak *mob spesifik ini* (misalnya, untuk pembaruan posisi, kematian). ID tipe digunakan untuk mencari statistiknya di mobs.json. Perubahan lantai di dungeon multi-level (terutama Avalonian<sup>55</sup>) mungkin memicu event spesifik atau urutan event (misalnya, pemain menghilang dari lantai lama, pemain muncul di lantai baru, serangkaian event kemunculan mob/peti baru). Ketidakmampuan pemindai sebelumnya untuk memindai beberapa lantai<sup>79</sup> menunjukkan kegagalan dalam mendeteksi atau menafsirkan transisi ini. README CleverCheat/albion-dungeon-scanner menyebutkan "Deteksi otomatis pintu masuk dungeon dan perubahan lantai melalui event Photon" sebagai *peningkatan potensial*<sup>2</sup>,



menyoroti pentingnya dan sifatnya yang tidak sepele.

## **B. Mendeteksi Kemunculan Mob, Tipe, dan Posisi**

Cari event seperti NewCharacter (<sup>81</sup> - meskipun <sup>83</sup> tampaknya lebih terkait parameter API), MobSpawn (<sup>54</sup>), atau event kustom Albion. Parameter dalam event ini harus mencakup:

- ID Instans Mob (unik untuk mob yang muncul itu).
- ID Tipe Mob (UniqueName, untuk dicari di mobs.json).
- Posisi (koordinat X, Y, Z).
- Kesehatan Saat Ini, Kesehatan Maksimum (mungkin datang dalam event pembaruan terpisah). Event penghilangan/kematian mob juga penting untuk menghapusnya dari daftar aktif pemindai. <sup>54</sup> menyediakan waktu *respawn* mob yang diamati komunitas, yang merupakan konteks tetapi tidak langsung dari paket. <sup>84</sup> mencantumkan mob sumber daya, beberapa di antaranya dapat muncul di dungeon.

Membedakan antara pemain, mob, dan entitas lain dalam event kemunculan adalah kunci. Parameter event atau kode event itu sendiri harus menyediakan diferensiasi ini. Event generik "EntityAppeared" tidak cukup. Pemindai perlu mengetahui apakah itu mob, peti, atau pemain lain. Informasi ini kemungkinan ada dalam *payload* event (misalnya, parameter tipe entitas). Posisi mob mungkin sering diperbarui melalui event "Move" atau "PositionUpdate" terpisah, yang merujuk pada ID Instans Mob. *OperationCode.Move* <sup>20</sup> adalah untuk gerakan yang diinisiasi pemain, tetapi mob akan memiliki pembaruan posisi yang didorong server. Mob tidak hanya muncul dan tetap statis. Pemindai perlu melacak pergerakan mereka untuk menampilkan posisi mereka saat ini secara akurat. Ini memerlukan identifikasi dan penguraian event pembaruan posisi.

## **C. Mendeteksi Kemunculan Peti, Tipe (Biasa, Langka, Epik, Legendaris, Buku), dan Status**

Mirip dengan mob, cari event kemunculan spesifik peti. Parameter harus mencakup:

- ID Instans Peti.
- ID Tipe Peti (UniqueName, untuk dicari di items.json). ID ini akan menentukan kelangkaannya (Hijau, Biru, Ungu, Emas/Legendaris <sup>6</sup>) dan apakah itu Peti Buku.
- Posisi. Event untuk perubahan status peti (misalnya, dibuka, menghilang) penting. <sup>47</sup> menjelaskan berbagai tipe peti dan warnanya. Pemindai athernot gagal mendeteksi beberapa peti, dan CleverCheat/albion-dungeon-scanner memiliki masalah dengan deteksi "ava chest". <sup>2947</sup> dan <sup>47</sup> mencantumkan berbagai tipe peti

termasuk peti Avalonian (Hijau, Biru, Ungu, Putih/Legendaris).

Perbedaan antara peti biasa dan "Peti Buku" <sup>6</sup> penting karena mereka menawarkan hadiah yang berbeda (buku ketenaran <sup>73</sup>). Pemindai harus dapat mengidentifikasi ini. UniqueName atau parameter tipe spesifik dalam event kemunculan akan membedakan peti buku dari peti *loot*. Informasi ini berharga bagi pengguna. Peti Avalonian memiliki set kelangkaannya sendiri, termasuk peti legendaris "Putih".<sup>47</sup> Kegagalan pemindai sebelumnya untuk mendeteksi "ava chest" <sup>29</sup> berarti ini mungkin menggunakan kode event atau parameter yang berbeda yang memerlukan identifikasi spesifik. Jika peti Avalonian menggunakan UniqueName yang berbeda atau ditandai oleh parameter event yang berbeda, logika pemindai harus disesuaikan untuk mengenalinya. Ini adalah area kunci untuk ditingkatkan dari proyek sebelumnya.

#### **D. Memetakan ID Entitas dari Paket ke Nama Data Game**

Ini melibatkan pembuatan sistem yang tangguh untuk:

1. Mengekstrak ID Tipe (UniqueName) dari paket event kemunculan untuk mob atau peti.
2. Menggunakan ID Tipe ini untuk mencari data statis entitas di *mobs.json* atau *items.json* yang telah diurai (misalnya, nama yang dilokalkan, statistik, faksi).
3. Menggunakan data lokalisasi (dari *JPCoDeCraft/AlbionLocalization* atau *LocalizedNames* di *ao-bin-dumps*) untuk mendapatkan nama tampilan. Pemetaan ini penting untuk menyajikan informasi yang ramah pengguna.

Menyimpan data game yang sering diakses (misalnya, definisi item dan mob) dalam memori dapat meningkatkan kinerja pemindai, menghindari pencarian file atau kueri basis data berulang kali. Terus-menerus membaca dari file JSON untuk setiap entitas yang muncul akan lambat. Cache dalam memori dari data *items.json* dan *mobs.json*, yang diindeks berdasarkan UniqueName, akan membuat pencarian jauh lebih cepat. Menangani kasus di mana ID dari paket tidak ada dalam file data game lokal (misalnya, karena file usang setelah pembaruan game) sangat penting untuk ketahanan. Pemindai harus menurun secara anggun (misalnya, menampilkan ID atau placeholder) daripada mogok. Jika tipe mob baru ditambahkan ke game dan *mobs.json* pemindai sudah usang, ia tidak akan menemukan definisi mob tersebut. Ini kemungkinan penyebab "error berkepanjangan" atau "akurasi tidak tepat." Pemindai memerlukan fallback untuk ID yang tidak dikenal.

## **VI. Teknik Pemindaian Tingkat Lanjut & Dungeon Avalonian**

Dungeon Avalonian menyajikan tantangan unik yang memerlukan pendekatan

pemindaian yang lebih canggih dibandingkan dungeon Solo atau Group standar.

## A. Wawasan Pemindaian Mendalam dari imjangkar/albibong dan M8-Babbage/avalon-dungeon-decoder

- **imjangkar/albibong**<sup>1</sup>:
  - Ini adalah pelacak kerusakan, ketenaran, dan *dungeon* lintas platform. README-nya menyebutkan penggunaan photon-packet-parser dengan modifikasi dan AlbionOnline-StatisticsAnalysis untuk kode event/op.
  - **Tugas Riset:** Analisis kode sumber albibong (Python/TypeScript) untuk memahami:
    - Bagaimana ia melacak entri dungeon, nama, ketenaran/perak yang diperoleh, dan mengaitkan snapshot DPS.
    - Penggunaan kode event/op Photon spesifiknya untuk aktivitas terkait dungeon.
    - "Modifikasi" yang dibuat pada photon-packet-parser – ini bisa jadi vital untuk penguraian paket Albion yang akurat.
    - Bagaimana ia menangani deteksi anggota party dan inisialisasi karakter ("ubah lokasi untuk menginisialisasi karakter").
  - Fitur pelacakan dungeon albibong<sup>1</sup> menunjukkan bahwa ia telah berhasil memecahkan beberapa masalah terkait identifikasi instans dungeon dan aktivitas pemain di dalamnya. Pendekatannya untuk mengorelasikan event dan data bisa sangat instruktif. Jika albibong dapat melacak "nama dungeon, ketenaran yang diperoleh, perak yang diperoleh, waktu mulai, durasi dungeon," ia pasti mengidentifikasi event masuk/keluar dungeon dengan benar dan mengaitkan aktivitas pemain dengan instans dungeon tertentu. Ini secara langsung relevan dengan tujuan pengguna.
- **M8-Babbage/avalon-dungeon-decoder**<sup>85</sup>:
  - Namanya sendiri menunjukkan bahwa ini khusus untuk dungeon Avalonian, menyiratkan bahwa mereka memerlukan logika penguraian khusus di luar pemindaian dungeon standar.
  - **Tugas Riset (jika sumber tersedia atau melalui pengetahuan komunitas):**
    - Data spesifik apa yang diuraikannya untuk dungeon Avalonian? (misalnya, tata letak kompleks, informasi multi-lantai, properti mob/peti unik, mekanisme gerbang<sup>55</sup>).
    - Mengapa dekoder *khusus* diperlukan? Apakah ia menangani struktur data unik, enkripsi (tidak mungkin mengingat sifat paket Albion umum), atau korelasi event multi-paket yang kompleks untuk status dungeon Avalonian?
  - Keberadaan avalon-dungeon-decoder (bahkan jika tidak dapat diakses)

sangat menyiratkan bahwa dungeon Avalonian menghadirkan tantangan penguraian unik yang tidak tercakup oleh logika pemindaian dungeon generik. Ini bisa jadi karena struktur peta yang lebih kompleks, tipe entitas unik, atau urutan event spesifik terkait mekanismenya (misalnya, gerbang energi<sup>55</sup>). Ini sejalan dengan laporan pengguna tentang pemindai yang kesulitan dengan konten Avalonian.<sup>79</sup> Pengembang tidak membuat alat khusus kecuali ada masalah spesifik dan kompleks yang perlu dipecahkan. Bagian "dekoder" menunjukkan bahwa data mentah dungeon Avalonian dari paket mungkin tidak dapat langsung digunakan tanpa pemrosesan atau interpretasi lebih lanjut, tidak seperti dungeon solo/grup yang lebih sederhana.

## B. Tantangan dan Data Spesifik untuk Dungeon Avalonian

- **Pemindaian Multi-Lantai:** Postingan forum<sup>80</sup> dan laporan masalah (<sup>79</sup> untuk CleverCheat/albion-dungeon-scanner) menunjukkan bahwa pemindai yang ada kesulitan dengan dungeon Avalonian multi-lantai, seringkali hanya menampilkan lantai pertama atau memberikan informasi yang salah. Dungeon Avalonian dapat memiliki 1-2 lantai.<sup>55</sup>
- **Mob dan Mekanisme Unik:** Dungeon Avalonian menampilkan mob unik (faksi Avalonians<sup>55</sup>) dengan kemampuan spesifik (misalnya, Purge Smoke dari Avalonian Acolyte, Bladestorm/Avalonian Might dari Knight<sup>55</sup>). Mereka juga memiliki mekanisme seperti gerbang energi di antara ruang bos.<sup>55</sup>
- **Kompleksitas Tata Letak:** Roads of Avalon dan Dungeon Avalonian dapat memiliki tata letak dan sistem portal yang rumit.<sup>65</sup> Meskipun pemindai berfokus pada *di dalam* dungeon, mekanisme masuk dan potensi struktur internal yang bervariasi relevan.
- **Sumber Data:**
  - Paket jaringan: Perlu mengidentifikasi event spesifik untuk kemampuan mob Avalonian, status gerbang, dan transisi lantai.
  - mobs.json: Harus berisi definisi untuk semua mob Avalonian.
  - items.json: Harus mendefinisikan peti Avalonian (termasuk legendaris "Putih"<sup>47</sup>).
  - Berpotensi file avalonianDungeons.json spesifik atau serupa di ao-bin-dumps untuk set ubin atau elemen struktural unik mereka.

Masalah "not detecting ava chest"<sup>29</sup> dan masalah dengan pemindaian multi-lantai<sup>79</sup> adalah rintangan utama untuk dungeon Avalonian. Riset harus memprioritaskan penemuan event dan parameter Photon yang secara andal menandakan tipe peti Avalonian dan perubahan lantai. Ini adalah titik kegagalan yang diketahui. Keberhasilan pemindai baru sangat bergantung pada penyelesaian masalah spesifik

ini. Ini mungkin melibatkan pencarian urutan event, nilai parameter spesifik dalam event generik, atau kode event yang sama sekali baru dan tidak terdokumentasi yang hanya digunakan dalam konten Avalonian. Saran oleh pengguna forum untuk membuat dungeon Avalonian memiliki hingga 5 lantai dan tidak ada peti di lantai pertama<sup>80</sup> untuk mematahkan pemindai menyiratkan bahwa pemindai saat ini mungkin mengandalkan informasi lantai pertama atau kesulitan dengan progresi yang lebih dalam. Ini menyoroti kebutuhan akan pemindai yang dapat memetakan sepenuhnya semua lantai yang dapat diakses.

### C. Memastikan Akurasi dan Menangani Kasus Tepi

Atasi masalah proyek sebelumnya: "akurasi tidak tepat," "miss scanning," "error berkepanjangan."

Strategi yang dapat diterapkan:

- **Manajemen Status:** Pertahankan representasi internal status dungeon yang akurat (mob, peti, posisi).
- **Konsistensi Akhir (*Eventual Consistency*):** Pahami bahwa data jaringan tiba dengan penundaan; tampilan pemindai sedikit di belakang status sebenarnya server.
- **Penanganan Error:** Tangani paket yang rusak, kode event yang tidak dikenal (catat dan lanjutkan), pemutusan koneksi, dan data game yang hilang secara tangguh.
- **Validasi Data:** Referensi silang informasi dari paket dengan data game statis jika memungkinkan.
- **Heuristik (Gunakan dengan Hati-hati):** Untuk situasi di mana data ambigu, heuristik dapat dipertimbangkan, tetapi juga dapat menyebabkan ketidakakuratan jika tidak dirancang dengan baik.
- **Pengujian:** Pengujian ketat dalam berbagai tipe dan skenario dungeon.

"Miss scanning" bisa disebabkan oleh paket yang hilang, pustaka *sniffing* gagal menangkap semua paket yang relevan, atau parser salah melewati event tertentu. Kehilangan paket adalah kenyataan dalam UDP. Meskipun Photon memiliki lapisan keandalan, *sniffer* itu sendiri mungkin kehilangan paket jika kelebihan beban atau tidak dikonfigurasi dengan benar. Logika parser juga perlu lengkap untuk event yang relevan. "Error berkepanjangan" menunjukkan pengecualian yang tidak ditangani atau status rusak yang tidak direset dengan benar. Ini menunjuk pada kebutuhan akan arsitektur perangkat lunak dan toleransi kesalahan yang lebih baik. Jika satu paket buruk atau event tak terduga merusak bagian inti pemindai atau merusak struktur data internalnya tanpa pemulihan, itu akan menyebabkan error terus-menerus. Desain modular dan manajemen status yang cermat adalah kuncinya.

## VII. Menyusun Struktur Aplikasi Pemindai (Pointer Riset Tingkat Tinggi)

Meskipun panduan ini berfokus pada pengumpulan informasi, pemahaman tentang bagaimana informasi tersebut akan digunakan dalam struktur aplikasi pemindai dapat membantu mengarahkan riset.

### A. Pertimbangan Arsitektural

- **Desain Modular:** Pisahkan komponen untuk:
  - Penangkapan Paket (berinteraksi dengan pustaka *sniffing*).
  - Deserialisasi Paket (parser protokol Photon).
  - Pemrosesan Event & Logika Game (menafsirkan event, memperbarui status).
  - Manajemen Data (mengakses/menyimpan data ao-bin-dumps, lokalisasi).
  - Antarmuka Pengguna. Modularitas ini membantu dalam pengujian, pemeliharaan, dan pembaruan bagian individual (misalnya, jika kode Photon berubah, hanya logika pemrosesan event yang mungkin perlu diperbarui). Pertimbangkan arsitektur berbasis event di mana event Photon yang diurai memicu pembaruan dalam model status game.

Desain monolitik lebih rentan terhadap masalah "error berkepanjangan." Jika semua logika digabungkan erat, error di satu bagian dapat meruntuhkan seluruh sistem atau merusak statusnya. Error terus-menerus proyek sebelumnya menunjukkan kurangnya isolasi kesalahan. Desain modular memungkinkan komponen gagal atau diperbarui secara independen, meningkatkan ketahanan keseluruhan. Kebutuhan untuk berpotensi memperbarui pemetaan kode Photon atau logika penguraian data game karena pembaruan game sangat mendukung desain di mana komponen ini mudah diganti atau dikonfigurasi. Pembaruan game tidak dapat dihindari. Arsitektur yang fleksibel memungkinkan pemindai beradaptasi lebih mudah, misalnya, dengan memuat definisi kode event dari file konfigurasi eksternal.

### B. Penyimpanan dan Manajemen Data untuk Informasi yang Dipindai

Pertimbangkan bagaimana pemindai akan menyimpan informasi tentang mob dan peti yang saat ini terdeteksi dalam sesi dungeon (struktur data dalam memori seperti kamus, daftar objek). Apakah akan menyimpan data apa pun di antara sesi? (misalnya, statistik, meskipun ini di luar pemindaian dasar). Pengambilan kueri dan pembaruan data dalam memori ini secara efisien saat event jaringan baru tiba juga penting.

Pilihan struktur data untuk entitas langsung akan memengaruhi kinerja, terutama di dungeon yang ramai dengan banyak mob. Jika dungeon memiliki ratusan entitas,

pencarian linier melalui daftar untuk pembaruan akan lambat. Menggunakan kamus yang dikunci oleh ID instans entitas untuk pencarian cepat lebih efisien. Manajemen memori menjadi perhatian jika pemindai berjalan untuk periode yang lama atau menghadapi dungeon dengan jumlah entitas unik yang sangat tinggi dari waktu ke waktu tanpa pembersihan entitas yang telah hilang (*despawn*) dengan benar. Kegagalan untuk menghapus entitas yang telah hilang dari daftar aktif dapat menyebabkan pembengkakan memori dan penurunan kinerja dari waktu ke waktu, yang berpotensi berkontribusi pada "error berkepanjangan."

### **C. Antarmuka Pengguna/Output (Serupa dengan theonlywitcher/Albion-Dungeon-Checker)**

Meskipun <sup>87</sup> (theonlywitcher/Albion-Dungeon-Checker) tidak dapat diakses, fungsionalitasnya dapat disimpulkan sebagai menampilkan daftar atau peta peti dan mob yang terdeteksi, tipe mereka, dan mungkin lokasi di dalam dungeon saat ini. pypye/albion-dungeon-checker <sup>6</sup> menyediakan output konsol sederhana: "LEGENDARY = Gold Chest," dll., mencantumkan mob dan medan. Ini adalah bentuk output dasar. CleverCheat/albion-dungeon-scanner <sup>2</sup> adalah alat baris perintah (pdm run scan) yang diperbarui saat dijalankan ulang pada perubahan lantai.

Pertimbangan riset meliputi bagaimana menyajikan data dengan jelas (tampilan daftar, overlay peta sederhana jika diizinkan secara etis dan layak secara teknis tanpa modifikasi game langsung), bagaimana menangani pembaruan saat entitas baru terdeteksi atau yang sudah ada berubah status/hilang, dan pembaruan otomatis vs. manual (seperti pemindai CleverCheat yang perlu dijalankan ulang pada perubahan lantai <sup>2</sup>).

Pengguna menginginkan output "serupa dengan theonlywitcher/Albion-Dungeon-Checker." Riset harus melibatkan pencarian diskusi komunitas, video (<sup>88</sup> - meskipun ini umum atau gameplay, bukan demo UI spesifik dari *checker* tersebut), atau *fork* yang mungkin menggambarkan UI-nya. Jika tidak ditemukan, pypye/albion-dungeon-checker <sup>6</sup> berfungsi sebagai format output yang lebih sederhana dan diketahui. Alat tolok ukur <sup>87</sup> adalah kotak hitam. Riset perlu menemukan proksi atau mendefinisikan serangkaian fitur UI yang masuk akal berdasarkan apa yang ditawarkan alat serupa dan apa yang layak. Overlay umumnya melanggar Persyaratan Layanan jika memberikan "manfaat langsung".<sup>12</sup> Jendela terpisah yang menampilkan informasi lebih aman. Kepatuhan Persyaratan Layanan sangat penting. Tampilan eksternal cenderung tidak ditandai daripada overlay dalam game.



## VIII. Praktik Terbaik, Legalitas, dan Pengembangan di Masa Depan

Pengembangan alat bantu untuk game online harus selalu mempertimbangkan aspek etika, legalitas, dan keberlanjutan jangka panjang.

### A. Mematuhi Persyaratan Layanan Albion Online

- **Prinsip Utama:** *Packet sniffing* untuk *analisis* dan *pengumpulan informasi* umumnya ditoleransi jika tidak mengotomatiskan tindakan atau memberikan keuntungan yang tidak adil.<sup>12</sup>
- **Yang Dilarang:**
  - Memodifikasi klien game atau paket.
  - Gameplay otomatis (bot).
  - Overlay yang memberikan keuntungan tempur/pengumpulan langsung (misalnya, radar yang menunjukkan entitas di luar visibilitas klien<sup>12</sup>).
  - Alat untuk pemetaan Roads of Avalon menggunakan data portal secara eksplisit dilarang pada satu titik.<sup>12</sup> Meskipun pemindai ini berfokus pada *interior dungeon*, preseden untuk membatasi alat berdasarkan data yang diendus ada.
- Pemindai harus menyajikan informasi yang secara teoretis dapat diamati sendiri oleh pemain dengan menjelajahi dungeon, tetapi diagregasi dan disajikan dengan nyaman.

Batas antara "alat informasi" dan "keuntungan tidak adil" bisa kabur dan tunduk pada interpretasi Sandbox Interactive (SBI).<sup>12</sup> Merancang alat agar murni informasional dan menghindari fitur apa pun yang mengotomatiskan keputusan pemain atau mengungkapkan hal-hal yang belum dikirim ke klien untuk dirender adalah pendekatan teraman. Sejarah AO-Radar<sup>12</sup> yang dilarang menunjukkan bahwa jika sebuah alat mengungkapkan terlalu banyak (misalnya, pemain/sumber daya di luar jangkauan visual tetapi dalam jangkauan jaringan), itu akan dianggap ilegal. Pemindai dungeon hanya boleh melaporkan entitas yang secara eksplisit telah diberitahukan server kepada klien untuk area saat ini. Fakta bahwa SBI telah membuat pernyataan tentang *packet sniffing*<sup>12</sup> menunjukkan bahwa mereka mengetahui alat-alat ini. Pengguna harus siap menghadapi potensi perubahan kebijakan.

### B. Strategi untuk Memelihara Pemindai di Tengah Pembaruan Game

Pembaruan game adalah alasan utama pemindai rusak (misalnya, "Not Working after Foundations Update" 29).

Strategi yang dapat diterapkan:

- Buat pemetaan kode event/op Photon mudah diperbarui (misalnya, file konfigurasi eksternal).
- Pantau repositori komunitas untuk data Albion Online (kode Photon, ao-bin-dumps) untuk pembaruan.
- Implementasikan pencatatan untuk tipe/kode paket yang tidak dikenal atau tidak ditangani untuk membantu dalam debugging setelah patch.
- Miliki proses untuk memverifikasi ulang kode dan struktur data setelah pembaruan game besar.

Sebagian besar pemeliharaan akan melibatkan pembaruan definisi kode dan logika penguraian data agar sesuai dengan perubahan dalam klien game. Ini bukan proyek "atur dan lupakan." Sifat dinamis game online memerlukan pemeliharaan berkelanjutan. Arsitektur modular yang dibahas dalam VII.A akan sangat membantu dalam hal ini, karena perubahan dapat dilokalkan ke komponen tertentu. Jika kode event berubah, hanya modul pemrosesan event yang perlu diperbarui, bukan seluruh aplikasi, jika dirancang dengan baik.

### **C. Potensi Peningkatan di Masa Depan (Di Luar Cakupan Riset Awal tetapi Baik untuk Konteks)**

- UI yang lebih canggih (misalnya, peta grafis lantai saat ini).
- Pelacakan statistik (misalnya, frekuensi tipe peti tertentu).
- Peringatan untuk mob/peti berharga tertentu.
- Integrasi dengan data pasar untuk perkiraan nilai *loot* (meskipun ini menambah kompleksitas dan bergantung pada API lain seperti Albion Online Data Project <sup>7</sup>).

Fokus pada persyaratan inti (deteksi mob/peti yang akurat) terlebih dahulu sangat penting. Peningkatan dapat datang kemudian. Kegagalan proyek pengguna sebelumnya terjadi pada hal-hal fundamental. Memperbaikinya adalah prioritas untuk pemindai baru. Peningkatan apa pun juga harus dipertimbangkan terhadap Persyaratan Layanan. Misalnya, peringatan untuk peti legendaris kemungkinan besar tidak masalah, tetapi mengarahkan pemain secara otomatis ke sana tidak akan diizinkan.

## **IX. Kesimpulan dan Rekomendasi**

Pengembangan pemindai dungeon Albion Online yang akurat dan andal adalah tugas yang kompleks namun dapat dicapai dengan pendekatan riset yang sistematis dan pemahaman mendalam tentang mekanisme game. Kegagalan proyek sebelumnya, seperti yang diuraikan oleh pengguna, kemungkinan besar berasal dari kombinasi masalah dalam penangkapan paket jaringan, deserialisasi protokol Photon yang tidak

tepat, kurangnya sinkronisasi dengan data game statis yang diperbarui, dan penanganan error yang kurang tangguh, terutama untuk konten yang lebih kompleks seperti dungeon Avalonian.

Riset yang diuraikan dalam panduan ini menekankan pentingnya:

1. **Fondasi yang Kuat:** Memilih bahasa pemrograman yang tepat (Python atau C# dengan pertimbangan matang), menyiapkan lingkungan pengembangan, dan menguasai alat analisis jaringan seperti Wireshark dan Npcap/libpcap. Mengatasi masalah fundamental "paket network tidak ditemukan" adalah prioritas utama.
2. **Pemahaman Protokol Photon:** Melakukan dekonstruksi Protocol16 secara cermat, mengidentifikasi OperationCode dan EventCode yang relevan (dengan validasi berkelanjutan terhadap sumber komunitas seperti Oblu/AlbionTracker dan Triky313/AlbionOnline-StatisticsAnalysis), dan menguasai deserialisasi tipe data Photon. Pemanfaatan pustaka photon-packet-parser yang ada, terutama versi yang telah dimodifikasi dan terbukti berfungsi seperti yang mungkin digunakan oleh imjangkar/albibong, dapat mengakselerasi pengembangan, tetapi harus diverifikasi secara menyeluruh.
3. **Integrasi Data Game Statis:** Memanfaatkan ao-bin-dumps secara efektif untuk mendapatkan data item, mob, dan dunia game. Memahami struktur file JSON (misalnya, items.json, mobs.json, randomizeddungeons.json) dan bagaimana UniqueName digunakan untuk menghubungkan data jaringan dengan data statis sangat penting. Data lokalisasi dari sumber seperti JPCodeCraft/AlbionLocalization atau bidang LocalizedNames dalam *dump* juga krusial untuk presentasi yang ramah pengguna.
4. **Penanganan Spesifik Dungeon Avalonian:** Memberikan perhatian khusus pada tantangan yang ditimbulkan oleh dungeon Avalonian, termasuk pemindaian multi-lantai dan identifikasi mob/peti unik mereka. Wawasan dari proyek seperti M8-Babbage/avalon-dungeon-decoder (jika dapat diakses atau dipahami melalui sumber sekunder) akan sangat berharga.
5. **Desain yang Tangguh dan Dapat Dipelihara:** Mengadopsi arsitektur modular untuk memfasilitasi pembaruan dan pemeliharaan sebagai respons terhadap patch game. Implementasikan penanganan error yang komprehensif dan manajemen status yang cermat untuk menghindari "error berkepanjangan" dan "akurasi tidak tepat."

## Rekomendasi Utama:

- **Prioritaskan Validasi Kode Jaringan:** Jangan hanya mengandalkan daftar kode yang ada. Lakukan verifikasi aktif menggunakan Wireshark dan gameplay untuk memastikan akurasi, terutama setelah pembaruan game.

- **Fokus pada Penanganan Error:** Implementasikan mekanisme penanganan error yang kuat di setiap tahap, mulai dari penangkapan paket hingga presentasi data, untuk mengatasi masalah seperti paket yang hilang atau rusak, kode yang tidak dikenal, dan data game yang usang.
- **Iterasi dan Uji Secara Menyeluruh:** Mulai dengan fungsionalitas inti untuk tipe dungeon yang lebih sederhana (Solo/Group) untuk memvalidasi pendekatan dasar sebelum menangani kompleksitas dungeon Avalonian. Lakukan pengujian ekstensif dalam berbagai skenario.
- **Patuhi Persyaratan Layanan:** Rancang pemindai sebagai alat informasional murni untuk menghindari pelanggaran Persyaratan Layanan Albion Online. Hindari fitur yang dapat dianggap memberikan keuntungan tidak adil atau mengotomatiskan gameplay.
- **Berkontribusi pada Komunitas (Opsional):** Jika kode atau data baru yang signifikan ditemukan selama riset, pertimbangkan untuk membagikannya kembali ke komunitas untuk membantu orang lain dan meningkatkan kualitas alat pihak ketiga secara keseluruhan.

Dengan mengikuti panduan riset ini dan menerapkan praktik pengembangan perangkat lunak yang baik, pengguna memiliki peluang besar untuk berhasil membangun pemindai dungeon Albion Online generasi berikutnya yang secara signifikan lebih unggul dari upaya sebelumnya dan memberikan nilai nyata bagi komunitas pemain.

## Works cited

1. imjangkar/albibong: A cross-platform Albion Online ... - GitHub, accessed May 31, 2025, <https://github.com/imjangkar/albibong>
2. CleverCheat/albion-dungeon-scanner: Simple Albion ... - GitHub, accessed May 31, 2025, <https://github.com/CleverCheat/albion-dungeon-scanner>
3. Realtime - .NET Platform SDK | Photon Engine - Fusion 2 Introduction, accessed May 31, 2025, <https://doc.photonengine.com/realtime/current/reference/dotnet-sdk>
4. albion-dungeon-checker/albion-tools.csproj at main · Oxpye/albion ..., accessed May 31, 2025, <https://github.com/pypye/albion-dungeon-checker/blob/main/albion-tools.csproj>
5. albion-dungeon-checker/Program.cs at main · Oxpye/albion ... - GitHub, accessed May 31, 2025, <https://github.com/pypye/albion-dungeon-checker/blob/main/Program.cs>
6. Dungeon Chest and Boss Checker for Albion Online - GitHub, accessed May 31, 2025, <https://github.com/pypye/albion-dungeon-checker>
7. JPCodeCraft/AlbionDataAvalonia - GitHub, accessed May 31, 2025, <https://github.com/JPCodeCraft/AlbionDataAvalonia>

8. santiac89/photon-packet-parser: A library to parse Photon ... - GitHub, accessed May 31, 2025, <https://github.com/santiac89/photon-packet-parser>
9. photon-packet-parser - PyPI, accessed May 31, 2025, <https://pypi.org/project/photon-packet-parser/>
10. SharpPcap - A Packet Capture Framework for .NET - CodeProject, accessed May 31, 2025, <https://www.codeproject.com/Articles/12458/SharpPcap-A-Packet-Capture-Framework-for-NET>
11. NScanner/SharpPcap/docs/SharpPcap.html at master · biboudis/NScanner - GitHub, accessed May 31, 2025, <https://github.com/biboudis/NScanner/blob/master/SharpPcap/docs/SharpPcap.html>
12. The State of Packet Inspection Tools : r/albiononline - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/albiononline/comments/ife1ou/the\\_state\\_of\\_packet\\_inspection\\_tools/](https://www.reddit.com/r/albiononline/comments/ife1ou/the_state_of_packet_inspection_tools/)
13. Retoon/AlbionOnline-StatisticsAnalysis-Ava-RH: A tool for easy reading of auction house data for the game AlbionOnline, and furthermore has been modified to support certain features designed to simplify the workflow for the Albion Avalonian Raid discord raid helpers and raid leaders. - GitHub, accessed May 31, 2025, <https://github.com/Retoon/AlbionOnline-StatisticsAnalysis-Ava-RH>
14. Triky313/AlbionOnline-StatisticsAnalysis: A tool with many ... - GitHub, accessed May 31, 2025, <https://github.com/Triky313/AlbionOnline-StatisticsAnalysis>
15. DocTi/albion-network: Provides convenient work with network events Albion Online. - GitHub, accessed May 31, 2025, <https://github.com/DocTi/albion-network>
16. 0xN0x/photon-packet-parser: A parser for photon's protocol16 packet - GitHub, accessed May 31, 2025, <https://github.com/0xN0x/photon-packet-parser>
17. Photon .NET Client API: ExitGames.Client.Photon.Protocol16 Class ..., accessed May 31, 2025, [https://doc-api.photonengine.com/en/dotnet/current/class\\_exit\\_games\\_1\\_1\\_client\\_1\\_1\\_photon\\_1\\_1\\_protocol16.html](https://doc-api.photonengine.com/en/dotnet/current/class_exit_games_1_1_client_1_1_photon_1_1_protocol16.html)
18. Releases · Triky313/AlbionOnline-StatisticsAnalysis - GitHub, accessed May 31, 2025, <https://github.com/Triky313/AlbionOnline-StatisticsAnalysis/releases>
19. mazurwikt0r/photon\_decode: Set of utilities for decoding messages generated by photon framework. - GitHub, accessed May 31, 2025, [https://github.com/mazurwikt0r/photon\\_decode](https://github.com/mazurwikt0r/photon_decode)
20. Packet tools for Albion Online - GitHub Gist, accessed May 31, 2025, <https://gist.github.com/Jadd/29b8b3cf8a82577ccad7d852a1d06ff6>
21. AlbionTracker/Albion.Common/Photon/OperationCodes.cs at master · Oblu/AlbionTracker · GitHub, accessed May 31, 2025, <https://github.com/Oblu/AlbionTracker/blob/master/Albion.Common/Photon/OperationCodes.cs>
22. OperationCode Class Reference - Photon Unity Networking, accessed May 31, 2025, [https://doc-api.photonengine.com/en/pun/v1/class\\_operation\\_code.html](https://doc-api.photonengine.com/en/pun/v1/class_operation_code.html)

23. Photon Unity Networking: EventCode Class Reference, accessed May 31, 2025, [https://doc-api.photonengine.com/en/pun/v1/class\\_event\\_code.html](https://doc-api.photonengine.com/en/pun/v1/class_event_code.html)
24. Photon .NET Client API: ExitGames.Client.Photon.EventData Class Reference, accessed May 31, 2025, [https://doc-api.photonengine.com/en/dotnet/current/class\\_exit\\_games\\_1\\_1\\_client\\_1\\_1\\_photon\\_1\\_1\\_event\\_data.html](https://doc-api.photonengine.com/en/dotnet/current/class_exit_games_1_1_client_1_1_photon_1_1_event_data.html)
25. Serialization in Photon - Pun 2 - Fusion 2 Introduction, accessed May 31, 2025, <https://doc.photonengine.com/pun/current/reference/serialization-in-photon>
26. github.com, accessed May 31, 2025, <https://github.com/Oblu/AlbionTracker/blob/master/Albion.Common/Photon/EventCodes.cs>
27. accessed January 1, 1970, <https://raw.githubusercontent.com/Oblu/AlbionTracker/master/Albion.Common/Photon/EventCodes.cs>
28. Issues · Triky313/AlbionOnline-StatisticsAnalysis - GitHub, accessed May 31, 2025, <https://github.com/Triky313/AlbionOnline-StatisticsAnalysis/issues>
29. Issues · CleverCheat/albion-dungeon-scanner · GitHub, accessed May 31, 2025, <https://github.com/CleverCheat/albion-dungeon-scanner/issues>
30. packetnet/PacketDotNet/UdpPacket.cs at master - GitHub, accessed May 31, 2025, <https://github.com/chmorgan/packetnet/blob/master/PacketDotNet/UdpPacket.cs>
31. photon-packet-parser - piwheels, accessed May 31, 2025, <https://www.piwheels.org/project/photon-packet-parser/>
32. Santiago Carullo santiac89 - GitHub, accessed May 31, 2025, <https://github.com/santiac89>
33. accessed January 1, 1970, <https://github.com/Selivaniuk/albionDataServer>
34. AoData - GitHub, accessed May 31, 2025, <https://github.com/ao-data>
35. broderickhyman/ao-bin-dumps: Albion Online Binary File ... - GitHub, accessed May 31, 2025, <https://github.com/broderickhyman/ao-bin-dumps>
36. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps>
37. Working on a Web-Marketplace Project.... need data dumps! : r/albiononline - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/albiononline/comments/6smyl4/working\\_on\\_a\\_webmarketplace\\_project\\_need\\_data/](https://www.reddit.com/r/albiononline/comments/6smyl4/working_on_a_webmarketplace_project_need_data/)
38. List of Items in JSON, XML or CSV format - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/45925-List-of-Items-in-JSON-XML-or-CSV-format/>
39. Is Albion Online friendly to 3rd partys who want to make Wiki's, databases, guides and other stuff? : r/albiononline - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/albiononline/comments/112huau/is\\_albion\\_online\\_friendly\\_to\\_3rd\\_partys\\_who\\_want/](https://www.reddit.com/r/albiononline/comments/112huau/is_albion_online_friendly_to_3rd_partys_who_want/)
40. broderickhyman/albiondata-bin-dumper - GitHub, accessed May 31, 2025, <https://github.com/broderickhyman/albiondata-bin-dumper>
41. broderickhyman/ao-id-extractor: The tool takes the Items.bin and world.bin file



- and Exports the ItemID's and LocationID's to the output file. - GitHub, accessed May 31, 2025, <https://github.com/broderickhyman/ao-id-extractor>
42. accessed January 1, 1970, <https://github.com/ao-data/ao-id-extractor>
43. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps/blob/master/items.json>
44. accessed January 1, 1970, <https://raw.githubusercontent.com/ao-data/ao-bin-dumps/master/formatted/items.json>
45. raw.githubusercontent.com, accessed May 31, 2025, <https://raw.githubusercontent.com/Fraser-Greenlee/my-huggingface-datasets/master/data/python-lines/test.json>
46. API | The Albion Online Data Project, accessed May 31, 2025, <https://www.albion-online-data.com/api-site/api.html>
47. Chests - Albion Online Wiki, accessed May 31, 2025, <https://wiki.albiononline.com/wiki/Chests>
48. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps/blob/master/mobs.json>
49. accessed January 1, 1970, <https://raw.githubusercontent.com/ao-data/ao-bin-dumps/master/formatted/mobs.json>
50. deep-code-search/pytorch/data/github/vocab.tokens.json at master, accessed May 31, 2025, <https://github.com/guxd/deep-code-search/blob/master/pytorch/data/github/vocab.tokens.json>
51. licenses/test/all.json at master - GitHub, accessed May 31, 2025, <https://github.com/3rd-Eden/licenses/blob/master/test/all.json>
52. raw.githubusercontent.com, accessed May 31, 2025, <https://raw.githubusercontent.com/NixOS/nixpkgs/nixos-21.11/pkgs/top-level/all-packages.nix>
53. uBlock Origin Filters - GitHub, accessed May 31, 2025, <https://raw.githubusercontent.com/ublockorigin/uassets/master/filters/filters.txt>
54. Does anyone have a guide of mobs' respawn times? : r/albiononline - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/albiononline/comments/18lofmk/does\\_anyone\\_have\\_a\\_guide\\_of\\_mobs\\_respawn\\_times/](https://www.reddit.com/r/albiononline/comments/18lofmk/does_anyone_have_a_guide_of_mobs_respawn_times/)
55. Avalonian Dungeons - Albion Online Wiki, accessed May 31, 2025, [https://wiki.albiononline.com/wiki/Avalonian\\_Dungeons](https://wiki.albiononline.com/wiki/Avalonian_Dungeons)
56. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps/blob/master/world.json>
57. accessed January 1, 1970, <https://raw.githubusercontent.com/ao-data/ao-bin-dumps/master/formatted/world.json>
58. Open World - Albion Online Wiki, accessed May 31, 2025, [https://wiki.albiononline.com/wiki/Open\\_World](https://wiki.albiononline.com/wiki/Open_World)
59. aaru-dps/Aaru: Aaru Data Preservation Suite - GitHub, accessed May 31, 2025,



- <https://github.com/aaru-dps/Aaru>
60. portaler-core/docs/devsetup.md at main · mawburn/portaler-core, accessed May 31, 2025, <https://github.com/mawburn/portaler-core/blob/main/docs/devsetup.md>
  61. paulstothard/helpful\_commands: Command-line tools, commands, and code snippets for performing routine data processing and bioinformatics tasks. - GitHub, accessed May 31, 2025, [https://github.com/paulstothard/helpful\\_commands](https://github.com/paulstothard/helpful_commands)
  62. raw.githubusercontent.com, accessed May 31, 2025, <https://raw.githubusercontent.com/rocks-moonscript-org/moonrocks-mirror/master/manifest-5.2>
  63. API Info | The Albion Online Data Project, accessed May 31, 2025, <https://www.albion-online-data.com/api-info/api-info.html>
  64. Albion Online Data API Introduction - Getting Market Prices | #203 - YouTube, accessed May 31, 2025, <https://www.youtube.com/watch?v=0sU0IYOFbJk>
  65. Roads of Avalon - Albion Online Wiki, accessed May 31, 2025, [https://wiki.albiononline.com/wiki/Roads\\_of\\_Avalon](https://wiki.albiononline.com/wiki/Roads_of_Avalon)
  66. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps/blob/master/randomizeddungeons.json>
  67. accessed January 1, 1970, <https://raw.githubusercontent.com/ao-data/ao-bin-dumps/master/formatted/randomizeddungeons.json>
  68. Guide needed: How to parse extremely complex JSONs the fastest way : r/rust - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/rust/comments/b56h7d/guide\\_needed\\_how\\_to\\_parse\\_extremely\\_complex\\_jsons/](https://www.reddit.com/r/rust/comments/b56h7d/guide_needed_how_to_parse_extremely_complex_jsons/)
  69. Best practices for parsing dynamic/unstructured JSON? : r/swift - Reddit, accessed May 31, 2025, [https://www.reddit.com/r/swift/comments/wyly56/best\\_practices\\_for\\_parsing\\_dynamicunstructured/](https://www.reddit.com/r/swift/comments/wyly56/best_practices_for_parsing_dynamicunstructured/)
  70. accessed January 1, 1970, <https://github.com/ao-data/ao-bin-dumps/blob/master/corrupteddungeons.json>
  71. accessed January 1, 1970, <https://raw.githubusercontent.com/ao-data/ao-bin-dumps/master/formatted/corrupteddungeons.json>
  72. Dungeon - Albion Online Wiki, accessed May 31, 2025, <https://wiki.albiononline.com/wiki/Dungeon>
  73. Book Chests.... - Feedback & Suggestions - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/206683-Book-Chests/>
  74. github.com, accessed May 31, 2025, <https://github.com/JPCoDeCraft/AlbionLocalization>
  75. Community Localization - Feedback & Suggestions - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/49556-Community-Localization/>

76. l18N of XML documents - localization - Stack Overflow, accessed May 31, 2025, <https://stackoverflow.com/questions/692789/l18n-of-xml-documents>
77. TMX file: How to streamline translations and localizations - Smartcat, accessed May 31, 2025, <https://www.smartcat.com/blog/tmx-file/>
78. TMX Files and Format - Transifex Help Center, accessed May 31, 2025, <https://help.transifex.com/en/articles/6838724-tmx-files-and-format>
79. Software not work :( · Issue #7 · CleverCheat/albion-dungeon-scanner ..., accessed May 31, 2025, <https://github.com/CleverCheat/albion-dungeon-scanner/issues/7>
80. large dungeon/avalonia dungeon chest scanner, rmt need fix - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/211661-large-dungeon-avalonia-dungeon-chest-scanner-rmt-need-fix/>
81. Albion Online New Character Tutorial - YouTube, accessed May 31, 2025, <https://www.youtube.com/watch?v=rCOkv8tBTC0>
82. The Gamer Guide - Issue 21 - Issuu, accessed May 31, 2025, [https://issuu.com/thegamingguide/docs/issuetwentyone\\_august-september2024](https://issuu.com/thegamingguide/docs/issuetwentyone_august-september2024)
83. Update on API - New Endpoints - News Archives - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/131017-Update-on-API-New-Endpoints/>
84. Resource Mobs - Albion Online Wiki, accessed May 31, 2025, [https://wiki.albiononline.com/wiki/Resource\\_Mobs](https://wiki.albiononline.com/wiki/Resource_Mobs)
85. accessed January 1, 1970, <https://github.com/M8-Babbage/avalon-dungeon-decoder>
86. Avalonians - Albion Online Wiki, accessed May 31, 2025, <https://wiki.albiononline.com/wiki/Avalonians>
87. accessed January 1, 1970, <https://github.com/theonlywitcher/Albion-Dungeon-Checker>
88. How to Find Dungeons in Albion Online - YouTube, accessed May 31, 2025, <https://www.youtube.com/watch?v=juJZXHEDinY>
89. Inspection of Group Dungeons! Solo Open World! Albion Online - YouTube, accessed May 31, 2025, <https://www.youtube.com/watch?v=oQkQVRaCwgU>
90. 8.4 Solo Dungeons in 2024 Worth It? Albion Online - YouTube, accessed May 31, 2025, <https://www.youtube.com/watch?v=AMbC9x8XvFk>
91. Packet sniffing - General Questions & Discussions - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/63298-Packet-sniffing/>
92. Albion Market API Encryption - Feedback & Suggestions, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/206875-Albion-Market-API-Encryption/>
93. Cooperative Economic Gameplay - Feedback & Suggestions - Albion Online Forum, accessed May 31, 2025, <https://forum.albiononline.com/index.php/Thread/206918-Cooperative-Economic-Gameplay/>
94. Max W4RPWISH - GitHub, accessed May 31, 2025, <https://github.com/W4RPWISH>

95. accessed January 1, 1970,  
[https://github.com/W4RPWISH/Albion\\_Op-EvCodes\\_Checker](https://github.com/W4RPWISH/Albion_Op-EvCodes_Checker)