SHIFTGLITCH

COGNITIVE EXPLOIT MANUAL

THE COGNITIVE EXPLOIT MANUAL

10 battle-tested cognitive protocols that rewire how you absorb,
retain, and deploy knowledge. For operatives serious about advancing beyond NPC status.

// CLASSIFICATION: OPEN SOURCE — SHARE FREELY //

Compiled by ShiftGlitch Systems — shiftglitch.com

// TABLE OF CONTENTS //

EXPLOIT INDEX

001 **MEMORY PALACE**
Method of Loci Protocol

002 **ACTIVE RECALL**
Testing Effect Protocol

003 **SPACED REPETITION**
Distributed Practice Algorithm

004 **CHUNKING**
Pattern Compression Protocol

005 **INTERLEAVING**
Chaos Training Protocol

006 **FEYNMAN PROTOCOL**
Teach-It Exploit

007 **DUAL CODING**
Visual-Verbal Fusion Protocol

008 **MIND MAPPING**
Concept Architecture Protocol

009 **SHADOW PROTOCOL**
Elaborative Interrogation Exploit

010 **MISSION DEBRIEF**
Metacognition Protocol

Introduction p.3
Rank Progression Guide p.14
Final Transmission p.15

Deploy every exploit at shiftglitch.com

// INTRODUCTION //

YOU'VE BEEN JACKED IN.

Welcome, operative. You are holding intelligence that most people never find.

The education system gave you the wrong tools. You were told to re-read, to highlight, to cram the night before. These strategies feel productive — and produce almost nothing. The neuroscience has been clear for decades. Almost nobody teaches it.

ShiftGlitch is built on the science of how your brain actually works. Every feature — the flashcard system, the focus protocols, the rank progression — is engineered around the cognitive mechanisms that govern long-term memory, attention, and skill acquisition.

This manual gives you the protocols. Ten cognitive exploits, field-tested by researchers and practitioners, explained in plain operational terms. Each one will change how you work.

You do not have to use all of them at once. Start with Exploit 002: Active Recall. It is the highest-return protocol in existence. If you build only that habit, you will outperform 90% of your competition.

After that, layer in Exploit 003 (Spaced Repetition) and watch the compounding effect kick in.

The exploits in this manual are ordered from foundational to advanced. Read through them once. Then come back to the ones most relevant to your current mission.

The Mainframe is waiting.

// EXPLOIT 001

001

MEMORY PALACE

Method of Loci Protocol

> **"Convert any information into a navigable mental environment."**

// WHAT IT IS

The Memory Palace is one of the oldest cognitive exploits on record. You take a familiar physical space — your house, a route you walk every day — and you plant the information you need to retain at specific locations within it. Your brain evolved to remember places and spatial layouts. This exploit hijacks that survival system and repurposes it for knowledge storage.

// THE PROTOCOL

1. Select a physical location you know perfectly (your home, a regular commute).
2. Walk through it mentally and identify 10–15 fixed anchor points in sequence.
3. At each anchor point, place a vivid, bizarre, or emotional image representing the concept you need to remember.
4. To retrieve the information, mentally walk the route and "collect" each image.
5. Repeat the walk 3 times immediately after encoding, then again 24 hours later.

FIELD APPLICATION: Definitions, ordered sequences, vocabulary lists, historical timelines, formulas.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 002

002

ACTIVE RECALL

Testing Effect Protocol

> **"Force your brain to retrieve — do not re-read. Retrieval IS the learning."**

// WHAT IT IS

Re-reading a page feels productive but produces almost zero retention. Active recall forces your brain to retrieve information rather than passively recognise it. Every time you successfully retrieve something, the memory trace strengthens. Every time you fail, you discover exactly where to focus next. This is the single highest-ROI exploit available.

// THE PROTOCOL

1. Read or study a section once. Close the material.
2. Without looking, write, say aloud, or type everything you remember.
3. Compare your output to the source. Note exactly what you missed.
4. Repeat the recall attempt for missed items immediately.
5. Return for another recall round after 24 hours and again after 72 hours.

FIELD APPLICATION: Everything. This is the foundation all other exploits run on top of.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 003

003

SPACED REPETITION

Distributed Practice Algorithm

> **"Review at the exact moment before forgetting. Never review too early, never too late."**

// WHAT IT IS

Your brain consolidates memory during sleep and time gaps. Cramming works short-term but collapses within 72 hours. Spaced repetition uses an algorithm (built into ShiftGlitch's flashcard system) that schedules reviews at precisely calibrated intervals — 1 day, 3 days, 1 week, 2 weeks, 1 month — expanding the gap each time you succeed. The result is long-term retention with minimum review time.

// THE PROTOCOL

1. Create a flashcard for each discrete concept you need to retain.
2. Answer each card honestly — do not peek before attempting.
3. Rate your confidence: Easy (long delay), Hard (short delay), Again (immediate).
4. Trust the algorithm. Do your due cards every day, even when it's only a few.
5. Never skip a due-card session — consistency compounds dramatically.

FIELD APPLICATION: Vocabulary, formulae, definitions, concepts, facts, procedural steps.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 004

004

CHUNKING

Pattern Compression Protocol

> **"Compress complex systems into single coherent units your brain can handle in one move."**

// WHAT IT IS

Working memory holds roughly 4 items at once. Chunking defeats this limit by grouping related pieces of information into a single, labelled "chunk." Once a chunk is formed, your brain treats it as one unit — freeing up bandwidth for more. Chess grandmasters don't memorise individual pieces; they recognise patterns (chunks) built over thousands of hours. You can use the same mechanism deliberately.

// THE PROTOCOL

1. Identify related concepts that share a pattern, rule, or function.
2. Assign the group a single label or keyword.
3. Practise triggering the whole chunk from just the label.
4. Build nested chunks — groups of chunks — into a hierarchy.
5. Test that each chunk can be retrieved in under 2 seconds from its label alone.

FIELD APPLICATION: Complex systems, processes, multi-step protocols, category-heavy content.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 005

005

INTERLEAVING

Chaos Training Protocol

> **"Mix topics deliberately. Your brain learns faster when confused."**

// WHAT IT IS

Blocked practice (doing 20 problems of type A, then 20 of type B) feels smooth and productive. Interleaved practice (randomly mixing type A, B, and C) feels harder and more chaotic — but produces dramatically better retention and transfer. The confusion forces your brain to identify WHICH approach to use, not just how to execute it. This simulates real-world conditions where problems don't arrive pre-labelled.

// THE PROTOCOL

1. List all the topics or problem types you are currently working on.
2. Instead of completing one block fully, rotate through topics on a random schedule.
3. After a session, identify which topic transitions felt hardest — those are your weaknesses.
4. Use the discomfort as a signal. The more confused you feel, the more your brain is working.
5. Interleave at minimum 3 different topics per session for full effect.

FIELD APPLICATION: Problem-solving disciplines, mathematics, languages, procedural skills.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 006

006

FEYNMAN PROTOCOL

Teach-It Exploit

> **"Explain it like you're teaching it. Gaps in your explanation = gaps in your understanding."**

// WHAT IT IS

Richard Feynman, Nobel Prize physicist, had a rule: if you cannot explain a concept in simple terms to someone who knows nothing about it, you do not understand it. Every technical word or vague phrase you rely on is a flag pointing to a hollow spot in your knowledge map. This exploit forces those hollow spots into the open where they can be addressed.

// THE PROTOCOL

1. Take a concept you believe you understand. Take a blank page.
2. Explain the concept as if teaching it to a complete beginner — no jargon.
3. Every time you reach for a technical term, replace it with a plain-language description.
4. Identify every point where you hesitated, struggled, or went vague.
5. Return to source material for exactly those gaps. Repeat until no gaps remain.

FIELD APPLICATION: Any concept requiring deep comprehension, not just memorisation.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 007

007

DUAL CODING

Visual-Verbal Fusion Protocol

> **"Encode information in two systems simultaneously — verbal and visual."**

// WHAT IT IS

Information processed through both visual and verbal channels creates two separate memory traces that reinforce each other. A diagram with labels is more resilient than either a diagram alone or a text description alone. When one trace weakens, the other provides retrieval support. Dual coding also forces you to re-represent information in a new format, which deepens processing.

// THE PROTOCOL

1. After studying text-based material, draw a diagram, flowchart, or map from memory.
2. After studying a diagram, write a full verbal explanation of what it shows.
3. Do not copy — translate. The re-representation is where the learning happens.
4. Build a personal visual library of diagrams for your most complex concepts.
5. When revising, alternate between verbal recall (close book, write it out) and visual recall (draw the map).

FIELD APPLICATION: Processes, systems, cause-and-effect chains, anatomical structures, historical sequences.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 008

MIND MAPPING

Concept Architecture Protocol

> **"Externalise your knowledge network. See the connections you didn't know existed."**

// WHAT IT IS

Knowledge is not a list. It is a web of connected nodes. A mind map makes that web visible. The process of building one forces you to identify: what connects to what, what the central concept is, what dependencies exist, and where the blank spaces are. The map itself is less important than the act of constructing it — which is where the deep processing occurs.

// THE PROTOCOL

1. Start with the central concept in the middle of a blank page.
2. Branch outward to major sub-concepts. Use single keywords per node.
3. Branch again from each sub-concept to details and examples.
4. Draw connections between nodes on different branches where they relate.
5. Rebuild the map from memory 24 hours later and compare to the original.

FIELD APPLICATION: Complex topics with many interconnected components, revision overviews, essay planning.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 009

009

SHADOW PROTOCOL

Elaborative Interrogation Exploit

> **"Ask WHY until there is nothing left to ask. Surface-level understanding collapses under pressure."**

// WHAT IT IS

Most knowledge is stored as isolated facts. Elaborative interrogation links each fact to the network of reasons and mechanisms that explain it. When you know WHY something is true, you can reconstruct it from first principles even if the direct memory fades. Facts connected to explanations are far more durable than facts stored in isolation.

// THE PROTOCOL

1. For any fact or rule you need to know, ask: "Why is this true?"
2. For the answer to that question, ask "Why?" again.
3. Continue until you reach bedrock — a foundational principle with no deeper why.
4. Connect this chain to something you already know well.
5. When reviewing, recall the chain, not just the surface fact.

FIELD APPLICATION: Science, law, mathematics, any domain where understanding > memorisation.

!' Deploy this exploit at shiftglitch.com

// EXPLOIT 010

010

MISSION DEBRIEF

Metacognition Protocol

> **"Review your own performance after every session. The debrief is where real growth happens."**

// WHAT IT IS

Most operatives finish a session and move on. High performers spend 5–10 minutes debriefing: what worked, what didn't, what was harder than expected, what new questions emerged. This metacognitive habit turns experience into learning about learning — progressively improving not just your knowledge, but the efficiency of your entire knowledge acquisition system.

// THE PROTOCOL

1. At the end of every session, spend 5 minutes writing a debrief.
2. Record: What did I actually absorb today? What confused me? What needs more depth?
3. Note which exploits felt effective and which felt like wasted effort.
4. Identify what you will do differently in the next session.
5. Review past debriefs before starting a new session to maintain continuity.

FIELD APPLICATION: Every session, every topic, every operative. Non-optional.

!' Deploy this exploit at shiftglitch.com

// RANK PROGRESSION //

THE ESCAPE ROUTE

RANK 00

NPC

Starting rank. You are jacked in but not yet aware of your situation.

RANK 01

SCRIPT KIDDIE

Complete 5 focus sessions. Run 10 active recall rounds. Build your first flashcard deck (20+ cards).

RANK 02

GLITCH TECH

Achieve a 7-day focus streak. Complete 3 Diagnostic missions. Deploy 3 different cognitive exploits.

RANK 03

NETRUNNER

Complete a Mission Debrief for every session for 14 consecutive days. Maintain 80%+ flashcard success rate over 30 days.

RANK 04

SYSTEM ADMIN

Complete a full Escape Run. Demonstrate mastery across all 10 cognitive exploits. Operate at Netrunner level for 30 days.

Ranks are earned through sustained consistent effort. They cannot be purchased, bypassed, or simulated.

Start your rank climb — free — at shiftglitch.com

FINAL TRANSMISSION

You now hold more practical knowledge about how human memory works than most people will ever encounter. The question is not whether these exploits work — the research is conclusive. The question is whether you will deploy them.

Consistency is the only differentiator. A mediocre operative who shows up every day will outrun a brilliant one who doesn't.

The Mainframe is waiting. Jack in.

---

shiftglitch.com

Free to start. No credit card. No trial period.

Questions: admin@shiftglitch.com

// END TRANSMISSION — SHIFTGLITCH SYSTEMS — © 2026 LOURENS BREYTENBACH //