HACK

ISSUE TWO 2022

A celebration of digital counter culture

mr licka
open source frequencies

Cyber Security Labs
Why they are important?

Wireless Wizardry
DragonOS & other RF Shinanigans

SMARTCYBER solutions

SMART

solu

TCYBER
tions

Editoral rantings and ravings

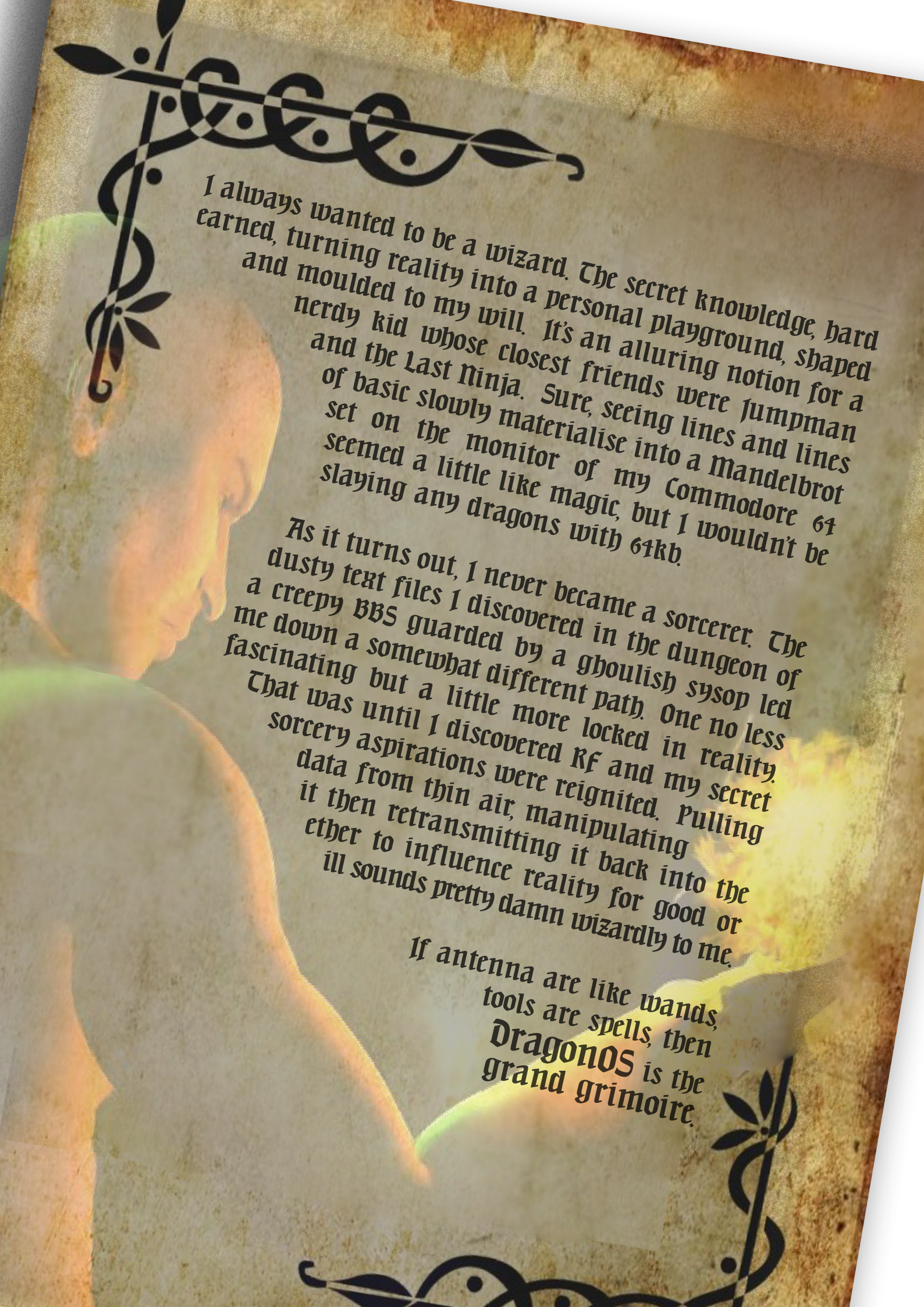
Pressure Heat Time
Heat Time
Pressure Time
Pressure Heat Time
Pressure
Pressure Time
Pressure Heat Time
Pressure

diamonds


JOKER

Wireless Wizardry

illuminations by
D8RH8R



I always wanted to be a wizard. The secret knowledge, hard earned, turning reality into a personal playground, shaped and moulded to my will. It's an alluring notion for a nerdy kid whose closest friends were Jumpman and the Last Ninja. Sure, seeing lines and lines of basic slowly materialise into a Mandelbrot set on the monitor of my Commodore 64 seemed a little like magic, but I wouldn't be slaying any dragons with 64kb.

As it turns out, I never became a sorcerer. The dusty text files I discovered in the dungeon of a creepy BBS guarded by a ghoulish sysop led me down a somewhat different path. One no less fascinating but a little more locked in reality. That was until I discovered RF and my secret sorcery aspirations were reignited. Pulling data from thin air, manipulating it then retransmitting it back into the ether to influence reality for good or ill sounds pretty damn wizardly to me.

If antenna are like wands, tools are spells, then **Dragonos** is the grand grimoire.

The DragonOS Origin Story

One day, when we are all old and gray, we will look back with amazement at all the changes that happened when coronavirus COVID-19 became a pandemic and the world went into voluntary quarantine, self-confinement, or lockdown, depending on where you lived. Most people spent much more time indoors at home. The morning commute changed almost overnight from dropping the kids off at school and a quick stop at Starbucks to letting the dog out the back door and a quick stop at the coffeemaker. In short, many found that they suddenly had more free time and fewer ways to use the newfound time productively, enjoyably, and safely. One such person was DragonOS Focal's developer, Cema Xecuter.

Cema Xecuter made the best of the situation and decided that the lockdown was more than a burden. It was an opportunity. It was a chance to learn more about RF, which was something he always wanted to do. He had also seen the enormous positive impact that Kali Linux had on offensive security, penetration testing, digital forensics, and white-hat hacking. Then inspiration struck, and he saw the fantastic potential in following the Kali Linux model, but with the focus on Software Defined Radio (SDR).

The plan was to create a Linux distribution with a comprehensive suite of pre-installed SDR software tools with support for many of the most popular and accessible SDR radios. No significant innovation comes without growing pains, and DragonOS is no different. DragonOS Focal follows in the footsteps of the first two DragonOS distributions DragonOS_10 and DragonOS_LTS, but Focal is the distribution that will stride into the future in lockstep with advancements in SDR hardware and software."

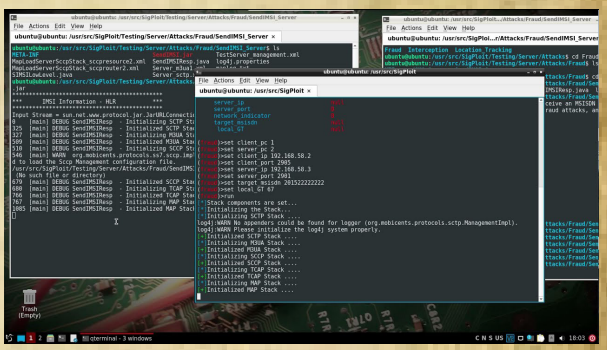
A conversation with Cema

I've been a fan of DragonOS for quite some time and recently had the opportunity to chat with the grand wizard himself.

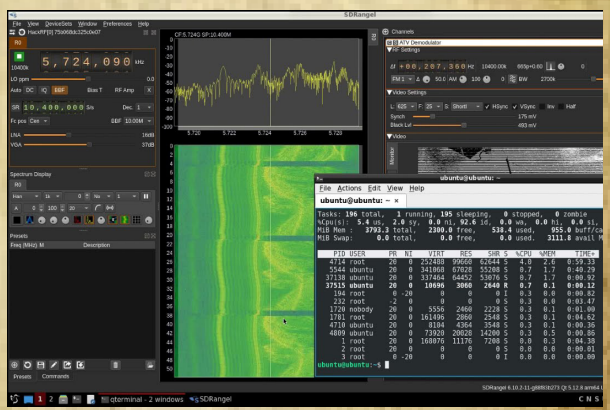
What motivated the first DragonOS?

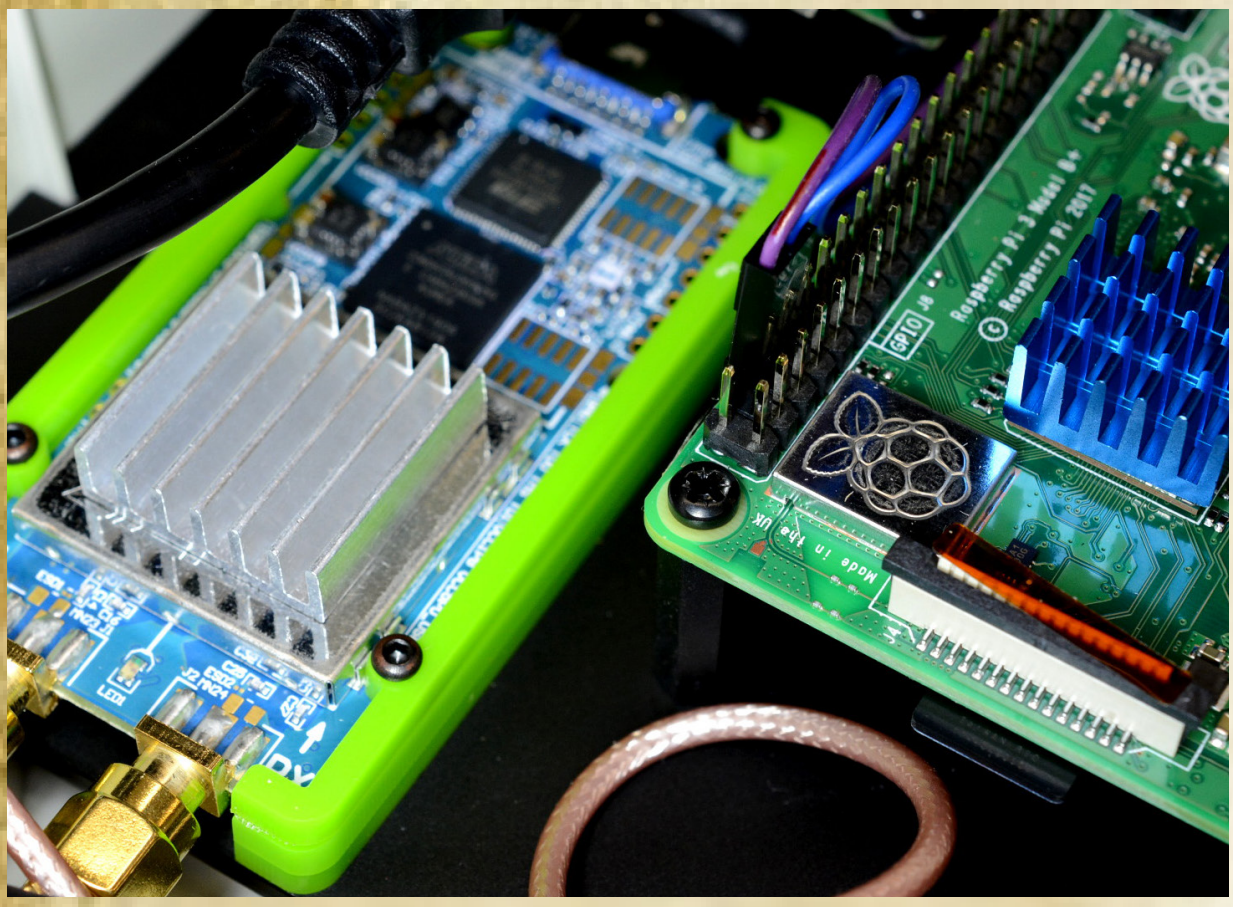
DragonOS was basically years in the making, I just didn't know it till Covid hit. Years and years ago I messed around with a project called Debian Enchilada, and not long after that I put together a live distribution for Zoneminder. Both distributions used a project called Remastersys, back then anyways. I've always enjoyed messing around with Linux, getting it to run on anything I could get my hands on, to include Xbox (Pay close attention to the background sound on my YouTube intros).

So fast forward to pre Covid and I was just getting into Software Defined Radios (SDR). I looked around and thought to myself, I want to make a distribution geared towards SDRs and I want it to encompass all the available SDRs I could get my hands on and I want it to have all the software I



SigPloit. Any awesome tool whether your an SDR nerd or not. Getting Gandalf as hell and casting mighty spells from the book of SCTP & SIGTRAN would also be nice.





could possibly jam into the ISO. As the origin story goes, I started out using Debian 10, later Ubuntu 18.04 and most recently 20.04.

What makes it different from other Signals Focused Distros?

I've only briefly tried SigintOS, so I really can't speak to other distros. I believe DragonOS is the only OS that's being put together meticulously by hand (really painful) in such a way that very unique applications, such as Crocodile Hunter (LTE Fake base station detection) or SDR4space (Javascript SDR Virtual Machine) is guaranteed to work out of the box.

In addition to the countless hours, over the course of about two years now, I not only refine and ensure everything is working - I make YouTube videos demonstrating exactly how to use the software that's included in DragonOS.

Also something unique about DragonOS is there's an aarch64 based release for the Pi4 that closely mirrors the desktop release. I can say it's the only image available online for the Pi4 that has all I've jammed inside the image.

What's coming up for DragonOS?

I would say the biggest thing coming up for DragonOS would be a 22.04 based release, however, that's still far down the road. The 20.04 based release still has the advantage of having all the software everyone spent so much time around the world during Covid working on being compatible with it. I would say here within the new few weeks, if not sooner, a new DragonOS Focal and Pi64 release will be available.

Keen to take DragonOS for a spin. Find the links for the latest versions below.

DragonOS Focal R24
https://sourceforge.net/projects/dragonos-focal/files/

DragonOS Pi64 R24
https://sourceforge.net/projects/dragonos-pi64/files/

Spell #281..
Bewitching of a Honda

Wand required:
HackRF (or BladeRF though we will only be covering the hackRF enchantment)

Spells used:
- •FCCID.io
- •HackRF One
- •Gqrx
- •GNURadio

FCCID.io is a site yui can research what frequency RF items operate at.

Honda Civic Attack:
2016-2020
(LX, EX, EX-L, Touring, Si, Type R)

•Key fob FCC ID: KR5V2X
•Key fob frequency: 433.215MHz
•Key fob modulation: FSK

The magic words:
hackrf_transfer -r filename.raw -f <]frequency in Hz]> # listen

Click the the fob to lock and unlock the car a few times.. (Or hide in the bushes and record you mates fob,)

Im not going to give the resat away but I can tell you that you wont need to worry about rolling codes. Don't know what that is? Here;s some homework.

• https://www.youtube.com/watch?v=1RipwqJG50c

Best SDRs for get started with DragonOS

HackRF One Software Defined Radio (SDR), ANT500 & SMA Antenna Adapter Bundle

HackRF One from Great Scott Gadgets is a Software Defined Radio peripheral capable of transmission or reception of radio signals from 1 MHz to 6 GHz. Designed to enable the test and development of modern and next-generation radio technologies, HackRF One is an open source hardware platform that can be used as a USB peripheral or programmed for stand-alone operation.
 1 MHz to 6 GHz operating frequency
 half-duplex SDR transceiver
 up to 20 million samples per second
 8-bit quadrature samples (8-bit I and 8-bit Q)
 compatible with GNU Radio, SDR#, and more
 software-configurable RX and TX gain and baseband filter
 software-controlled antenna port power (50 mA at 3.3 V)
 SMA female antenna connector
 SMA female clock input and output for synchronization
 convenient buttons for programming
 internal pin headers for expansion
 Hi-Speed USB 2.0
 USB-powered
 open source hardware

Ubertooth One SDR

Ubertooth One SDRThe Ubertooth One is an open-source 2.4 GHz wireless development platform suitable for Bluetooth experimentation. One thing that sets the Ubertooth apart from other Bluetooth development platforms is that it's capable of not only sending and receiving 2.4 GHz signals, but can also operate in monitor mode, monitoring Bluetooth traffic in real-time. This operating mode of the Ubertooth One has been present in low-cost WiFi modules for years and has found myriad uses in research, development, and security auditing but no such solution existed for the Bluetooth standard until now. Also, because it's a fully open-source platform (software and hardware), the schematics and code are readily available for all of your hacking needs.

YARD Stick One USB Transceiver & 915MHz Antenna

YARD (Yet Another Radio Dongle) Stick One can transmit or receive digital wireless signals at frequencies below 1 GHz. It uses the same radio circuit as the popular IM-Me. The radio functions that are possible by customizing IM-Me firmware are now at your fingertips when you attach YARD Stick One to a computer via USB.

Capabilities:
 half-duplex transmit and receive
 official operating frequencies: 300-348 MHz, 391-464 MHz, and 782-928 MHz
 unofficial operating frequencies: 281-361 MHz, 378-481 MHz, and 749-962 MHz
 modulations: ASK, OOK, GFSK, 2-FSK, 4-FSK, MSK
 data rates up to 500 kbps
 Full-Speed USB 2.0

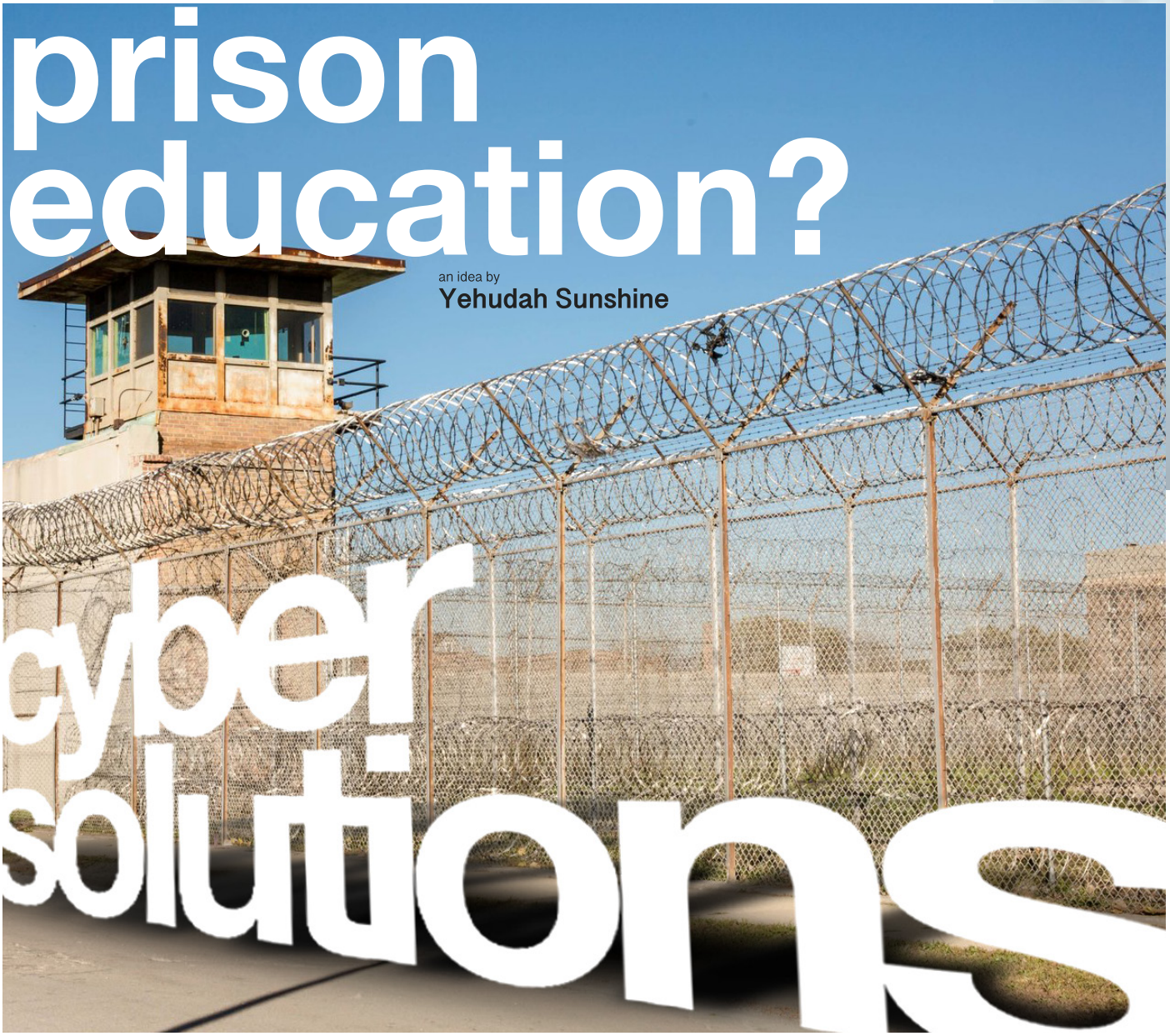
YARD Stick One comes with RfCat firmware installed, courtesy of atlas. RfCat allows you to control the wireless transceiver from an interactive Python shell or your own program running on your computer. YARD Stick One also has CC Bootloader installed, so you can upgrade RFCat or install your own firmware without any additional programming hardware. An antenna is not included. ANT500 is recommended as a starter antenna for YARD Stick One.

The DragonOS

out of the box

Acarsdec w/ rtlsdr support
Aircrack-ng 1.6
Airspy_ADSB
Apache2
Asterisk
Auto137
BladeRF ADSB w/ Dump1090 Mutability (/usr/src)
BladeRF-Wiphy (usr/src/wiphy-build)
Blue hydra
Boatbod op25
BTLE w/ hackrf (can be recompiled for bladeRF)
CalypsoBTS w/ firmware + tools
Cesium
Chirp-daily (python2)
Composable-SDR AppImage with SDRPlay support (/usr/src/Soapy_SDR_x86_64)
Crocodile Hunter (LimeSDR Mini support)
CubicSDR
DF-Aggregator w/ Offline capability
Direwolf
Dumphfdl
DumpVDL2
Esptool
FALCON
Fldigi
GNU Radio 3.8
Gpredict
GQRX
GQRX Scanner
GR-ADSB
GR-AIR-Modes
GR-AOA
GR-Correctiq
GR-DECT2
GR-DSD
GR-FHSS_Utils
GR-Foo
GR-Grnet
GR-GSM
GR-ieee802-11 w/ HackRF Sink TX Flowgraph
GR-ieee802-15-4
GR-IIO
Gr-Inspector (/usr/src)
GR-Iridium
GR-limesdr
GR-Lora
GR-Lora_SDR
GR-Mixalot
GR-NFC
GR-NRSC5
GR-NTSC-RC
GR-Paint38
GR-PDU_Utils
GR-RDS
GR-Sandia_Utils
GR-Satellites
GR-Smart_Meters
GR-Soapy
GR-Tempest
GR-Timing_Utils
Grgsm_scanner-GUI
GSMEvil2
HackTV GUI v2021-11-09
Ham2Mon by lordmorgul
IceCast2 (needs configured before starting)
IMSI-catcher
Inotify-tools
Inspectrum 0.2.3
Iridium-Toolkit
IridiumLive
JAERO
Js8call
JTDX
Kalibrate (HackRF)
Kismet
Kismet rest api
Kisstatic2mobile w/ latest kismet support
Kismon
Larry Tetra Kit e9f93618
LeanSDR/LeanDVB
Libacars
LibBladeRF 2.4.1 w/ xA5 support
LibhackRF/hackRF tools 2021.03.1
Libosmo-dsp
LimeSuite
Linrad
LiquidSoap
LTE-Cell-Scanner (v2 remains and supports RTLSDR
HackRF
BladeRF with CMake options)
LuaRadio v0.10.0 w/ examples
M17-Gnuradio
Meshtastic Python API 1.2.58(standalone)
Mirage (GitHub.com/RCayre/mirage)
MMDVM
Mmdvm-sdr by r4d10n
MMDVMHost by g4klx
Multimon-ng
Nmap
NOAA-Apt 1.3.1
NRSC5 decoder for RTLSDR
Nzyme
OP25 “Boatbod” (GNU Radio 3.8/Python3 testing /usr/src/op25/)
OpenWebRX 0.20.3
Osmo-bsc
Osmo-bts-trx
Osmo-hlr
Osmo-MGW
Osmo-msc
Osmo-NITB
Osmo-nitb-scripts (@NotPike)
Osmo-Sip-Connector
Osmo-trx-lms (LimeSDR support)
Osmo-trx-uhd
Osmocom-BB tools in /usr/src
Photonmap
Probequest
Pyadi-iio
PyRtlSDR
PySDR 2.0 (Guide to SDR and DSP using Python)
Qalculate
Qfits for use with sattools
QradioLink w/ MMDVM ability
QspectrumAnalyzer
Qsstv
Qt-DAB
RDF-Sim
Retrogram-RTLSDR
Retrogram-soapysdr
Reveng
RFcat
RFCrack
RFsoapyfile
RMSViewer
RSP TCP Server (SDRPlay support)
RTL_433
RTLSDR-Airband v4.0.1 (conf in /usr/src/)
RX_Tools
SatDump
SDR++ w/ server capability
SDR4space.lite w/ examples
SDRAngel
SDRReceiver
SDRTrunk
ShinySDR
SigDigger
Signal Server GUI w/ python3 virtual environment
Signal Server N90ZB w/ Web Interface by Dr. Bill Walker
Soapysdr modules
Sparrow-WiFi w/ FALCON tools + wpapcap2john
Splat!
SpyServer (usr/src/spyserver-linux-x64)
srsLTE-Sniffer (loop-catcher.sh in /usr/src/srsLTE-release_18_12/build/lib/examples)
srsRAN
Strf
Tetra-kit “screen2tetra.sh” script in /usr/src/tetra-kit/recorder/wav
Tetra-Kit-Player in /usr/src (needs npm installed)
Trunk-Recorder
Ubertooth 2020-12-R1
Umurmur
Universal Radio Hacker
WFView from source
wireguard
Wireshark
WSJT-X
Yate/YateBTS w/ BladeRF xA4 improvements
yellowShoes nrsc5 HD FM audio player
Zenmap

How can cybersecurity skills building revitalize vocational and prison education?



an idea by
Yehudah Sunshine

As of June 2022, conservative estimates gauge the global prison population at over 10.35 MILLION people. From petty criminals and low-level delinquents to murders and everything in between, the international criminal justice system manages and is financially responsible for the health, welfare, and basic needs of everyone who walks through their doors.

While in the past the prime focus of these institutions was to instill punishment and acknowledgment of their moral ills, increasingly corrections officials are turning to new avenues to impact the trajectory of their inmates.

As you might expect, there's no such thing as a free lunch (that is unless you're on the other side of the bars) so how will corrections officials provide a new path to prevent recidivism?

Cybersecurity training just might be the key to breaking the cycle of reoffending.

Who is impacted?

To many, prisons today appear to focus more on punishment rather than skills building. Between managing the massive populations, decaying infrastructure, and outdated mindsets, corrections facilities and officials often lack the essential support and educational opportunities to ensure their inmates achieve. As a direct result of these structural limitations, many prisoners fail to transcend their previous transgressions and reoffend.

To put the numbers into context. On average, the United States dept of corrections releases greater than 7 million people from jail and more than 600,000 people from prison annually. However, within 3 years of their release, 66% are rearrested and more than 50% are incarcerated again. Globally, re-arrest rates were between 26% and 60%, reconviction rates ranged from 20% to 63%, and reimprisonment rates varied from 14 to 45%.

Is cybersecurity education the solution?

Unable to hide from the sheer scale of global prison populations, their ballooning costs, and fading prospects of changing the hearts and minds of prisoners a crossroad has been reached.

On one side of the equation are the dated and ineffective educational programs proven to accept high recidivism without batting an eye.

On the other sits innovative vocational programs, skills-based training, and computer-focused courses aimed to revitalize the potential of prisoners about to regain their freedom.

In practice, the formula has been proven time and time again. The ability to succeed outside of prison is dependent on skills building, support structure, vocational education and training, and in the end the prospects of an honest livelihood.

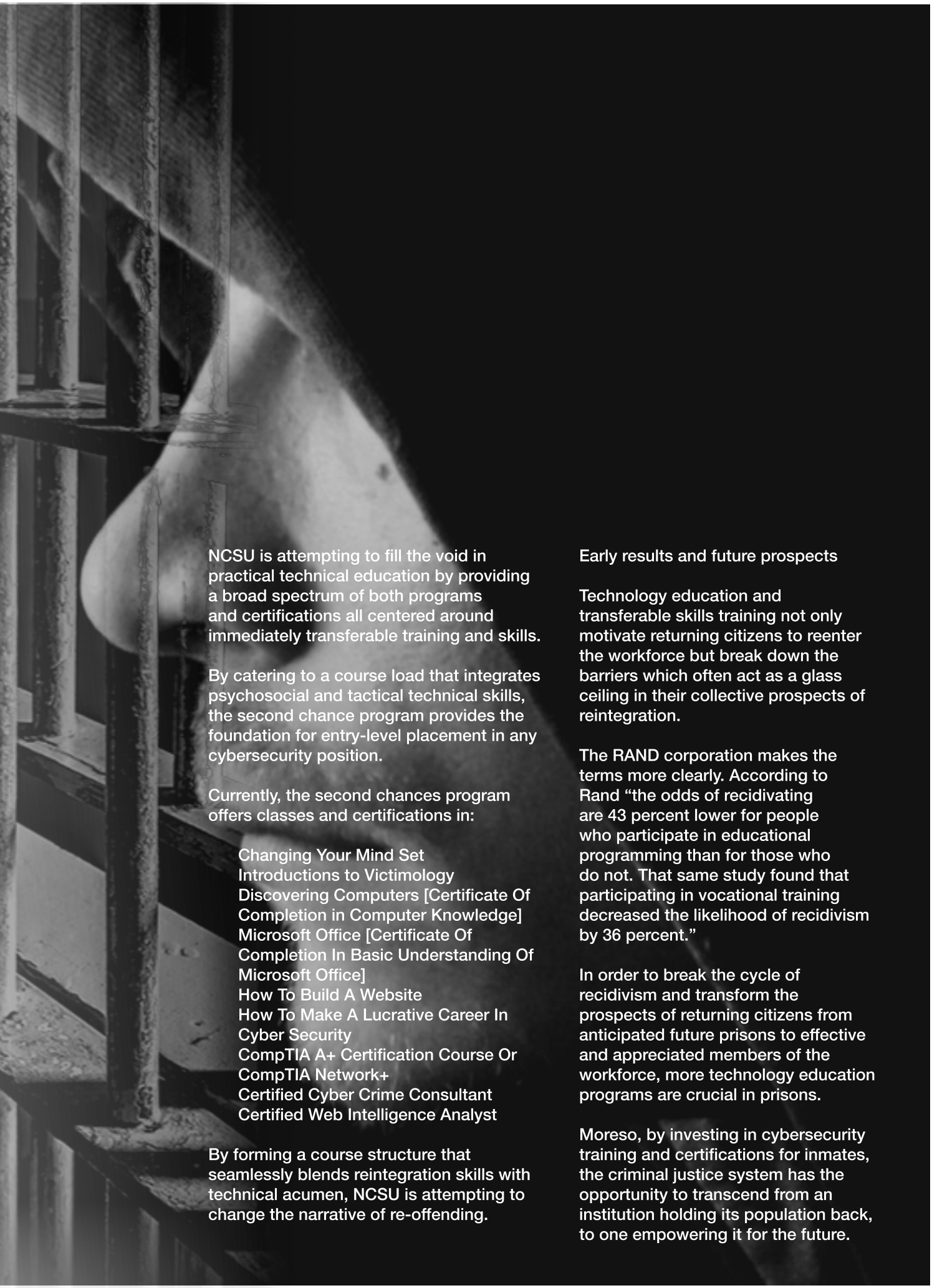
While some elements are obviously out of the purview of the criminal justice system, the ability to provide transferable certifications, technical skills, and technological literacy to the inmate population can often be the chief factor impacting recidivism.

From the Italian system enabling its prisons to pursue Cisco Networking certifications to the rise of NICCS's Second chance National Cybersecurity University, cybersecurity training is increasingly being turned to as a means to empower prisoners to build careers outside of crime.

Case Study: National Cybersecurity University Second Chance-Prison Reform

Currently, the National Cyber Security University is the only organization globally solely focused on educating rehabilitated inmates in the field of IT/cyber security. NCSU's core mission views cybersecurity as a way to empower prisoners for IT opportunities by supplying the tools and knowledge that will enable them gainful employment and an improved sense of self-worth.

Bringing together his diverse professional cyber know-how, intellectual fascination with history and culture, and eclectic academic background focusing on diplomacy and the cultures of Central Asia, Yehudah Sunshine keenly blends his deep understanding of the global tech ecosystem with a nuanced worldview of the underlying socio-economic and political forces which drive policy and impact innovation in the cyber sectors. Yehudah's current work focuses on how to create and enhance marketing strategies and cyber-driven thought leadership for Cyfluencer, the cybersecurity influencer platform trusted by vendors and cyber thought leaders. Sunshine has written and researched extensively within cybersecurity, the service sectors, international criminal accountability, Israel's economy, Israeli diplomatic inroads, Israeli innovation and technology, and Chinese economic policy.



NCSU is attempting to fill the void in practical technical education by providing a broad spectrum of both programs and certifications all centered around immediately transferable training and skills.

By catering to a course load that integrates psychosocial and tactical technical skills, the second chance program provides the foundation for entry-level placement in any cybersecurity position.

Currently, the second chances program offers classes and certifications in:

- Changing Your Mind Set
- Introductions to Victimology
- Discovering Computers [Certificate Of Completion in Computer Knowledge]
- Microsoft Office [Certificate Of Completion In Basic Understanding Of Microsoft Office]
- How To Build A Website
- How To Make A Lucrative Career In Cyber Security
- CompTIA A+ Certification Course Or CompTIA Network+
- Certified Cyber Crime Consultant
- Certified Web Intelligence Analyst

By forming a course structure that seamlessly blends reintegration skills with technical acumen, NCSU is attempting to change the narrative of re-offending.

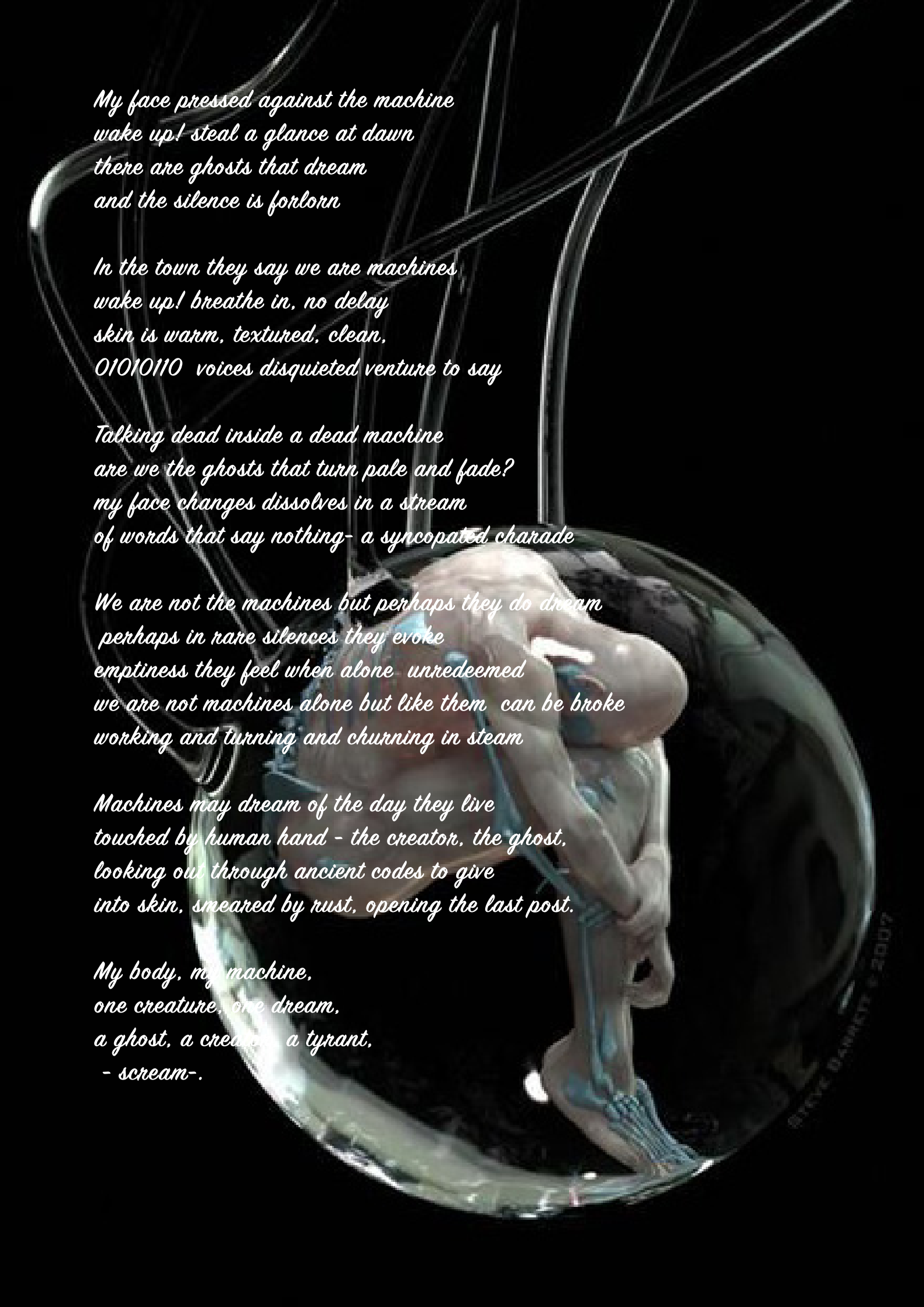
Early results and future prospects

Technology education and transferable skills training not only motivate returning citizens to reenter the workforce but break down the barriers which often act as a glass ceiling in their collective prospects of reintegration.

The RAND corporation makes the terms more clearly. According to Rand “the odds of recidivating are 43 percent lower for people who participate in educational programming than for those who do not. That same study found that participating in vocational training decreased the likelihood of recidivism by 36 percent.”

In order to break the cycle of recidivism and transform the prospects of returning citizens from anticipated future prisons to effective and appreciated members of the workforce, more technology education programs are crucial in prisons.

Moreso, by investing in cybersecurity training and certifications for inmates, the criminal justice system has the opportunity to transcend from an institution holding its population back, to one empowering it for the future.



My face pressed against the machine
wake up! steal a glance at dawn
there are ghosts that dream
and the silence is forlorn

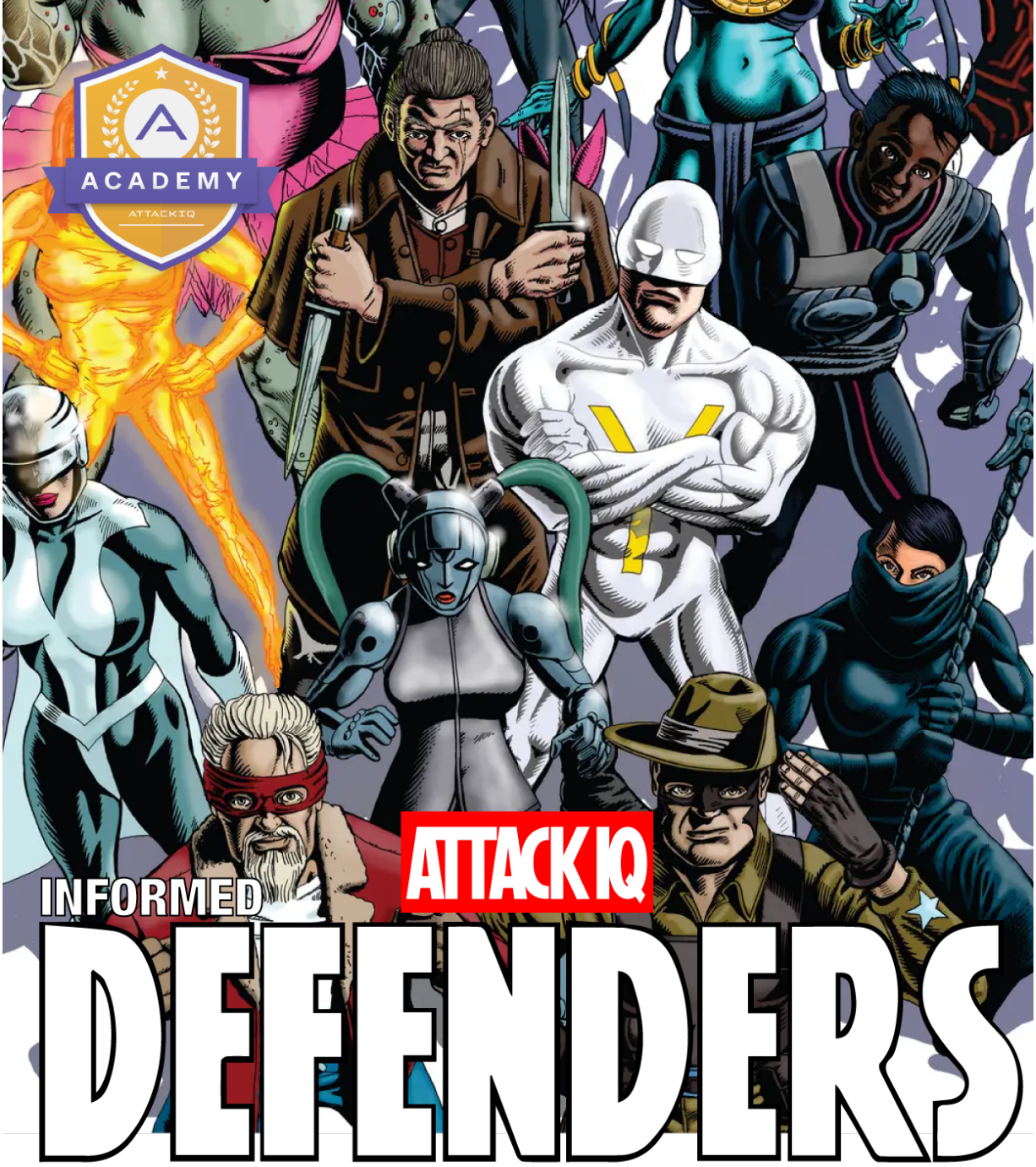
In the town they say we are machines
wake up! breathe in, no delay
skin is warm, textured, clean,
01010110 voices disquieted venture to say

Talking dead inside a dead machine
are we the ghosts that turn pale and fade?
my face changes dissolves in a stream
of words that say nothing- a syncopated charade

We are not the machines but perhaps they do dream
perhaps in rare silences they evoke
emptiness they feel when alone unredeemed
we are not machines alone but like them can be broke
working and turning and churning in steam

Machines may dream of the day they live
touched by human hand - the creator, the ghost,
looking out through ancient codes to give
into skin, smeared by rust, opening the last post.

My body, my machine,
one creature, one dream,
a ghost, a creator, a tyrant,
- scream-.



melbournecybersecuritymeetup

Friday 2nd August

Melbourne

venue tbc

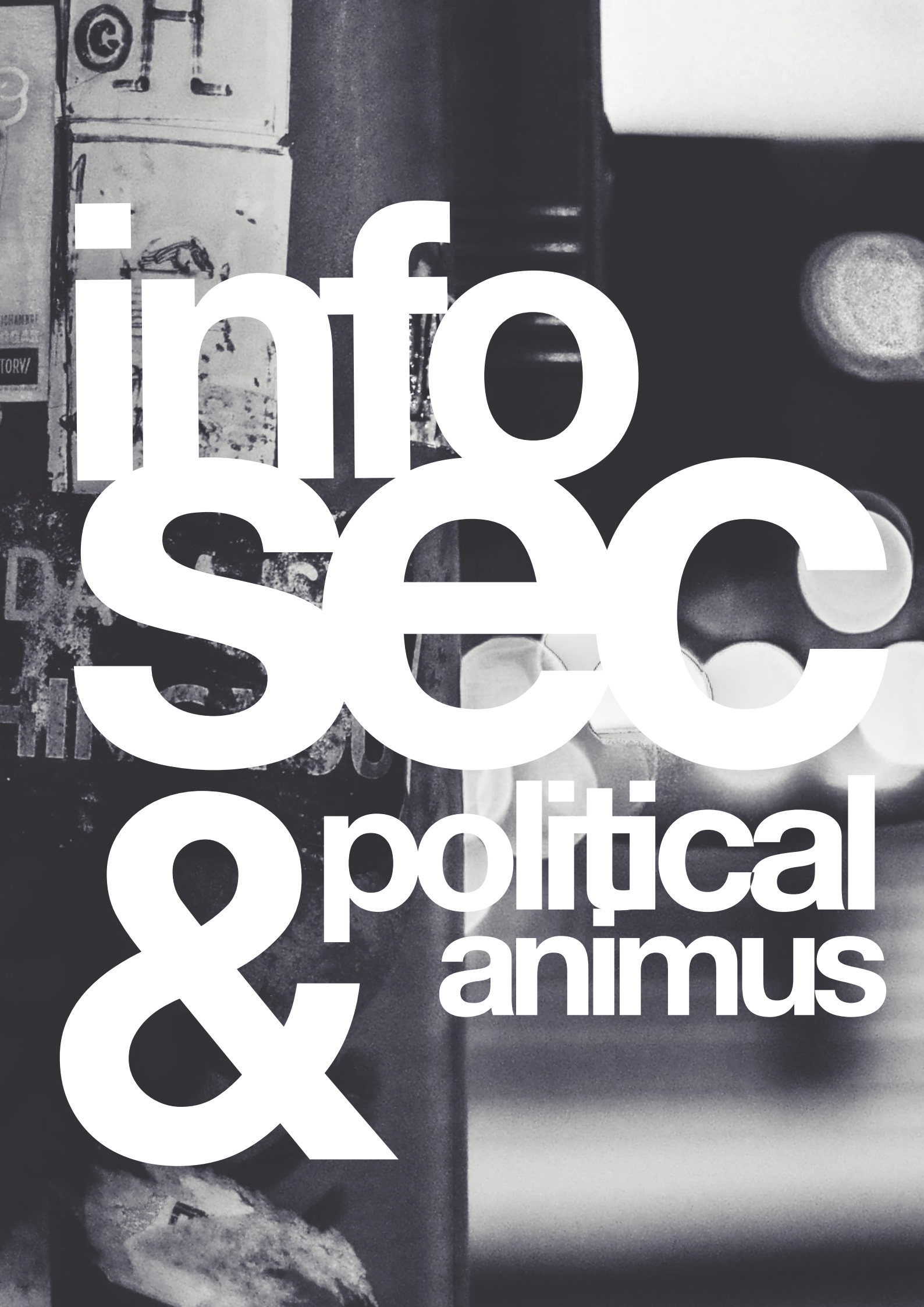
For more information check Linkedin or the Eventbrite page

Student to CISO - All are welcome

connect • collaborate • co-operate

HACK · SMARTCYBER · THE SAFER INTERNET PROJECT · Perfect Monday · ATTACKIQ INFORMED DEFENDERS CHAMPION

email melbournecybermeetup@protonmail.comfor more info


infosec
&
political
animus

the nexus between

Writen by Rio Whitehouse

“What happens when unstoppable forces collid with immovable objects?

A question asked many times and adapted for many situations, though interestingly not heard all that often in the InfoSec profession. If we reframe the question for InfoSec, we might arrive at a question like:

What happens when the duties of the job collide with personal political animus?

We all certainly have aspects of our jobs that we don’t completely agree with, and at times we’re all asked to discharge our duties in ways that offend our sensibilities, but in the throes of incident response or sanctioned red team engagements, how does one truly know where everyone stands?

Before getting into the weeds on this topic, I need to state that I’m in no way casting aspersions on anyone. I’m just a guy from a land down under, who, when asked, couldn’t help but wonder. There’s plenty of commentary around cybersecurity, and politics, but very little (possibly none) with the right blend of cybersecurity, political agendas, and workplace conflict.

Analysis of an Analyst’s Animus

Picture, if you will. You’re a SOC Analyst working for an MSSP (Managed Security Services Provider) who protects ESG companies from external and internal threats. The role is everything you could ask for at this point in your career, the work is fulfilling, and the company is aligned with your values and belief systems.

One fine Wednesday, you join the morning briefing to learn that one of your clients has been attacked overnight and Hive ransomware deployed across the client’s office network. It’s all hands on deck, duties are doled out and the incident response is firmly underway - you’re tasked with restoring selected SMB servers from backups.

While dutifully restoring the contents of the file servers and verifying the contents of the shares, you notice some metadata referencing a gas mining venture, of which the affected client is a significant stakeholder. This goes against both your own personal beliefs on climate action and renewable energy opportunites, as well as your employer’s ethos about helping ESG companies. What do you do from here? Do you set aside your beliefs and discharge your duty no matter how offensive the act may be, or do you disclose your position on the matter and take yourself off the incident response, putting a blemish on your record and potentially cutting off your career progression? Or... does an Elliot Alderson

monologue race through your mind, presenting a forbidden third option by way of a medley of cajoling, corruption and compromise?

Let's look at the same situation from the perspective of the SOC shift lead or the SOC Manager. While trying to control the inevitable chaos that washes over a department in the throes of disaster recovery, there's a corrosive notion in the heart and mind of an otherwise exemplary analyst. Can team leaders and managers account for such a situation? If so, how, and can it be done without stepping on freedom of political expression in the workplace, while at the same time enabling them to relegate those who might be better suited for other incident response scenarios?

What follows is general commentary on facets of Australian employment legislation and how it might apply to this scenario, and does not constitute legal advice in any form whatsoever. This is a complicated topic involving federal and state legislature, please seek independent legal advice in the jurisdiction that applies to you.

A reasonable first place to look might be federal law in your country. In Australia, we have a general implied freedom to express political views that has been acknowledged by the High Court of Australia[^1]. There are exceptions to this, such as using violence to express a political opinion, which falls under our criminal code, thus making us liable to criminal actions if we use violence in the course of political expression.

Australians also have some protections from political discrimination in the [Fair Work Act 2009 (Cth)](https://www.legislation.gov.au/Details/C2017C00323), that include political party memberships, civic commitments, and expressed political, socio-political or moral attitudes; however, again there are exceptions. If your views or opinions are deemed "unauthorised and inconsistent" with your role or the organisation's values, then an employer might take action on that basis. Legislation at the state level might offer some more distinct protections from political discrimination; this is certainly true in several Australian states (ACT, NT, Queensland, Tasmania and Victoria), but again there are exceptions and criteria that a situation would need to meet before being deemed as something an employer could take action against.

All of this is to say that even though employers could take steps to determine an employee's personal political animus, their hands may be tied by legislation up until the point where political expression is in direct conflict with organisation values, or if an act of violence occurs in the course of that political expression.

If (Civic_Expression == Cyber_Espionage)

In our scenario of a SOC Analyst wrestling with the angel and devil on either shoulder, examine for a moment what that third option might be:

Leak the project details. Shame the client for touting itself as an ethical organisation, while in the same breath choosing to be a slave to money, rather than remaining steadfast in the service of its mission. The client's ESG score will drop, they'll lose face as a result, and their stock price will tank.

A dangerous precedent to entertain, no doubt, not just because it fits squarely into the definition of cyber espionage[^2], but some may interpret this as an act of civic commitment. To some, it may seem like exactly the kind of exposé found on WikiLeaks and then covered by mainstream outlets. To others, it's a form of industrial espionage and charges should be brought forward.

Notwithstanding the legal ramifications for all involved, the negative publicity and damage control inside the MSSP would be catastrophic. The MSSP could forseeably have other clients terminate contracts as a result of the analyst's actions, ultimately cutting off the company at the knees and potentially send them hurtling towards bankruptcy. A business razed, an entire staff out of a job, and a blow to the InfoSec industry as a whole, all because of the socio-political slant of one person with the right access, tipped at just the right moment...sounds like the perfect social engineering exploit, doesn't it?

Why Bring This Up At All?

Surely there's a point to all of this, right? Yes, there is.

With the recent [disclosure](https://hackerone.com/reports/1622449) by HackerOne, whereby an employee accessed security reports for personal gain by using them to effectively double-dip on vulnerabilities already disclosed on the platform, this may be the impetus to new pre-employment screening for potential sources of conflict that could present in day-to-day operations or incident response. Or, it could be the stalking horse used to justify in-depth interview questions and aptitude tests designed to capture many data points, colouring right on the lines of what is and isn't acceptable to ask in interviews and on the job.

I'm hopeful that the incident at HackerOne doesn't become a test case for political screening beyond that of obtaining security clearances, but with the current political climate, I suppose anything is possible.

[^1]: [The Implied Freedom of Political Communication: The State of the Law Post Coleman and Mulholland](http://www.austlii.edu.au/au/journals/JCULawRw/2005/5.txt/cgi-bin/download.cgi/download/au/journals/JCULRev/2005/5.pdf [^2]: [What Is Cyber Espionage](https://www.crowdstrike.com/cybersecurity-101/cyberattacks/cyber-espionage/)

Making it easy for cybersecurity vendors and professionals to share quality content, ideas and insights

CYFLUENCER

Cyfluencer, the influencer marketing platform built exclusively for the cybersecurity community, is offering a free content, SEO & Ads audit, as well as 50 free clicks on the Cyfluencer Platform! To learn more contact Yehudah@Cyfluencer.com

We know there are so many threats out there, and people need solutions and awareness so they can protect themselves. But, so much marketing in the industry preys on fear, vulnerabilities, and panic.

As white hat marketers, we're not here to deceive, terrorize or catastrophize. By helping credible members of the industry make a bigger impact, we hope to use marketing for the good.

Vendors need a way to get their product out there, with integrity. Professionals want to share their deep expertise and promote meaningful dialogue. By connecting these two parties, we help vendors expand their reach, professionals to increase their impact, and the world to be a safer place.

Cyfluencer Offers

- **Influencer content sharing platform**
- **Cybersecurity Content Consultations**
- **Influencer Led Content Creation for your next Webinar or Whitepaper**
- **SEO and Ads Audits for Cybersecurity**

To Learn more about how Cyfluencer can enhance your Cybersecurity Marketing strategy visit our website or send us an email:

cyfluencer.com  yehudah@cyfluencer.com

Cyber Security Labs

Why are they important?

Article by **David Lee** sks **DC CyberSec**

Today's new recruits in the cyber security job market are finding it increasingly more difficult to get that first foot in the door. Some of the job postings I've seen in the last couple of months are asking for experience of over 3 years for an entry level position, which makes you think, "how is it even possible to get a job in cyber security?". Before I get to techniques on how to overcome this experience "requirement", I would like to discuss the topic of experience a little further.

Experience:

If you've ever watched any of my videos on YouTube or read any of my blogs on cybersecguidance.com, you would have heard me say "experience is king". It's a truth that I have to repeat often, as a lot of student's fresh out of college find it hard to find a job when all they have to show is a degree in computer science (or similar). We need to rewind a few years back to around 2010, when "Cyber Security" as an industry within IT first sort of started to show its teeth as an area outside of systems, networks and software development. Endpoint management, Email security, Firewalls and Layer 1 network security were starting to become hot topics and SIEM's starting to push through to the small to medium business markets as an affordable product. At this time, these businesses didn't have dedicated Cyber Security professionals, instead having either Systems Administrators or Network Engineers, or sometimes both Systems and Network Administrators all bundled together into one. For the hiring managers at these organisations (especially in Australia), shifting their mentality of hiring outside of these more traditional spaces is still a bit hard to understand, especially when there are still so many Network Engineers and Systems Administrators wanting to break into the Cyber Security industry by utilizing their years of professional work experience.

So, imagine that you're fresh out of university - bright eyed and bushy tailed - you're looking to prove yourself against the masses! Don't stress, there's this magical form of experience that you can do from the comfort of your own home - Home Labs!

Home Labs:

Creating your own home lab is an absolutely vital part in showcasing your skills. The way to go about proving these skills works similar to how the STAR methodology (Situation Task Action Result) works, and would be relayed to a hiring manager in an interview. Start with what it is you're creating this lab for, the problem it solves, then explain exactly how it was created, for example; "I have a blue team lab for analysing packets in my home network so I have a better understanding of what is going in and out of my network. I achieved this by creating a virtual firewall appliance that sits underneath my router. The logs from here export to a logging machine which I use {insert log analysing tool here} to inspect the traffic in an easy to manage way. I have specific types of packets setup with automatic notifications so I know exactly when something dodgy is going on. With this, I am able to achieve a better understanding of what is happening on my network and potentially stop any malicious activities from occurring, and have so far stopped {insert how many and which type of attack here} from breaking through to my devices on the network."

This might sound very simple and while it can be achieved with free tools, it showcases the very basic front-line layer of a Security Operations Center (SOC). Pretty cool huh?

There are other types of home labs that you can create which relate directly to the job you're apply for, and showing these skills off as experience on your resume followed up with huge amounts of passion and pride on your part will help to push you ahead in an interview.


s
freq
wha
taught
eternal li
mr lic

open source ligencies

t mr licka me about fe.....

ka

Music is my medicine.

Music is my liberator, physician, the catalyst for catharsis and the light that guides me home when everything has been broken and I need a delicate thread to bind the essential parts of me together into a new constellation of self. Music speaks to me, whispers secrets I barely comprehend but feel, deep, deep, down in the hollow spaces inside my bones.

It dares me to open every pore and exhale into the world just beyond my skin, and consider the possibility of being more, and yet asking nothing but to fall into everything that I am, in the moment. I am not a trained musician and yet I have always experienced a sensual resonance with sound, enriched through training as a dancer. The theory of music may not be my native language, but the poetry of sound that infuses my body and calls it back to the wild, back to the remembrance of it's natural state, is known and understood at the centre of every cell within this fragile form.

Here at HVCK we have an insatiable curiousity about what happens at the interface between the human and the digital and how these bodies of flesh and blood and feeling commune with the world of machines. When I was given the opportunity to interview one of the most talented electronic musicians in Australia I did not overlook the possibility of exploring not only the way in

which machines are shaping the way we create art, but how our artistic engagement with machines may offer us new insight and a deeper sense of meaning, belonging and connection within the unremitting, unflinching, and unwavering technological onslaught of this increasingly digitalised world.

My conversation with Mr.Licka began with me writing a letter (via email) consisting of questions emerging from my own curiousity. As can only be expected of a true artist, his responses revealed him to be a human whose art emerges from within and yet is an echo of the vastness of the world outside, with all it's complexity and possibility.

Here it is, the reflections of the artist in his own words, raw, rich and unedited.

What/who were the primary influences in your discovery and exploration of music?

Originally it was pop music when I was around 4 or 5. I would spend hours in front of the TV watching anything that had music in it. I was really good at memorising TV jingles too, I could sing most of the ads on the television commercial breaks, which was hilarious to my parents, but also a great early introduction to melody and phrasing. I discovered glam rock in the late 80's and became obsessed with bands like Poison and Guns N Roses and learnt the value of showmanship and gravitated towards the drums which would go on to be my profession and allow me to travel all over the world performing. As I hit adolescence, my musical tastes became increasingly heavier and heavier, and I can safely say I was a metal kid until the age of about 36. That was the age I discovered outdoor dance parties and the associated culture and I haven't looked back since. I still enjoy heavy music, but my creative focus is now more on psychedelic electronic music.

Describe your inspirations/motivations. What drives you? What lights your fire?

I've always been a creative and driven to make music in different iterations. I always wonder where the drive comes from to be honest. Is it from a need to be adored or admired? Is it a more pure form of creation where it existed in me the whole time? Maybe it's a combination of both? Why are most artists a little troubled? Does adversity breed creation? I think in my case, it was a tumultuous relationship with my mother that pushed me into music so much. I think I took solace in my practice and found value in myself when I advanced musically. I often wonder if this is a familiar tale amongst other musicians. That was early on mind you, my mother and I are fantastic now and have a great relationship. What drives me now I think, is still finding great value in contributing art to the world, and seeing my growth as a producer through the whole process.

I'm curious about your transition from traditional percussion (drums) to electronic music.

What was the catalyst for this shift? What called you to explore the weaving of electronic sounds ?

It was quite a quick shift to be honest, and I give psychedelics most of the credit. Me and my band mates at the time were recording our first full length album for a major record label on the Gold Coast. The producer introduced me to DMT one night after we'd finished for the day. This was a light bulb moment for me and totally changed my world view. I couldn't believe that this experience was possible and how the whole world wasn't talking about it!!! I literally met disembodied entities (Aliens) who guided me through a super technological realm and infused me with knowledge though their telepathic language. It was EXTREMELY profound. This set about a series of discoveries in the real world which lead to my travelling to Peru to drink Ayahuasca with the Shamans. My musical tastes started to morph shortly after this as I was searching for music which represented the psychedelic experience or harmonised with it. I started attending outdoor dance parties and eventually realised that I wanted make this music too, as it was really resonating with my being. It was one moment in particular at Earthcore 2015 where I received a msg in my head saying "you're going to make this music". It was a very strange experience for me, but ultimately it has proven true.

Is your art making an extension of who you are?

For example, is your music an expression/ reflection of who are in any moment? An ephemeral impression of your experience.

It most definitely is! I find inspiration in nearly everything I experience. Movies, music, relationships etc. There's a butterfly effect happening at all times which can dictate your creative process. Even a small, seemingly insignificant event can influence your writing. We are at the mercy of these forces at all times, even when we're not in the creative process. Every single human on earth is steered by these subtle forces.

HVCK:arts

Always fat,
always fresh.

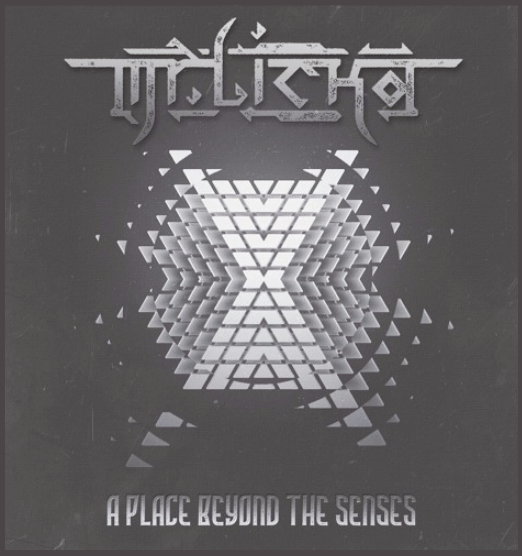
Choice Cuts

Dystracted
Silense (Mr Licka Remix)

The sunny coast never fails to bring the goods. This humble magazine, Mr licka himself and the groupd whos tune he's rerubbed. Thie whole compilation is worth checking out.

https://dystracted.bandcamp.com/track/silense-mr-licka-remix

Mr Licka
Limitless



From his debut offering on Universal Tribe Records, this tune some's up mr licka for me. You never know whats going to be around the next bend, but with out fail, you a lifted onto the back of another relentless groove. The funk you were just feeling already forgotten and replaced with rhythm and rapture.

https://soundcloud.com/mrlicka/limitless?utm_source=clipboard&utm_medium=text&utm_campaign=social_sharing

Can you describe your creative process?
Do you use other artistic/creative mediums/modalities to guide/inform/deepen your art making?
Do you start with a feeling, an impression, a desire, a beat?

I literally Just start tinkering with synths or trawling though samples until I come across something that inspires me. I liken the process to mining precious metals, in the fact that, there's a lot of stuff you don't need in the sample libraries or the synths, but occasionally you hit that pay dirt. Then you gotta flesh that nugget out and keep searching. The funny thing is, that each preceding nugget will dictate in a way, the things that come after it.

For example, if I use a certain synth patch or sample, that sound might not go with another sound because of a frequency clash or a timbral disparity. This means I have to then find sounds that harmonise with the already existing parts of the track. So in a way, the song starts to write itself based

Oh! Jupiter, our Father! If you would deliver men from all evils that oppress them,

Show them of what daemon they make use.

But take courage; the race of humans is divine.

Sacred nature reveals to them the most hidden mysteries.

If she impart to you her secrets, you will easily perform all the things which I have ordained thee.

And by the healing of your soul, you wilt deliver it from all evils, from all afflictions.

...when, after having deprived yourself of your mortal body, you arrived at the most pure Aither,

You shall be a God, immortal, incorruptible, and Death shall have no more dominion over you

(Pythagoras, trans.1707)

on decisions you made earlier on. It's crazy....all the options can sometimes feel overwhelming, as it's almost infinite the way things can generate, but you just have to trust your ears and experience and keep going.

When you see people responding to your music, ie through movement/dance does it shape the way you curate the sound in the moment? (A feedback loop, a co-creation)

I think you're referring to performing live here perhaps. All my sets are planned out and I don't really stray from the rehearsed set. One thing I will say about that, is when I write, I often have breaks where I'll play back the section of music I'm working on and dance around in the studio to see if it feels right to move to. This is a great way of gaining a different perspective on your music, by not listening to it staring at the screen and hearing it like a dancer would at the party. This has proven useful many times as a way to critically listen to the music, and make necessary changes.

Is the ritual/transformative/shamanistic quality of art something that you are curious about?

Absolutely!!!!

I realise the value of music in a psychedelic session, be it on a dance floor, or in a ceremonial setting. Music has the power to steer experiencers in a certain way and this is a very big responsibility. That's why I write the music I do. I write a more uplifting brand of Bush Prog because that's what I want to feel when I'm on psychedelics... uplifted and euphoric, not dark and brooding. But that's my vibe, some people might not resonate with that, and that's fine. There's plenty of other guys or girls that can fill the dark quota, it just isn't my thing.

What are your thoughts on machines and digital technology as artistic mediums?
Do you feel they deepen or subvert or connection to self, other, the earth?

Not at all!

We're only using machines to manipulate an already existing frequency spectrum. We're not creating new frequencies or musical notes. We still adhere to the laws set forth by Pythagoras in ancient greece. I understand that puritans and people who subscribe to the 432 hZ tuning standard will disagree as machines quite often rely on a 440 hZ tuning, but I offer this thought...

Back before there were any exact tuning mechanisms, there would have been only ear tuning, which is not exact, so there could have been ancient instruments tuned to a wide variety of different tunings based on the inaccuracy of the human ear. Does that mean that any instrument tuned slightly out of 432 hZ tuning back in the day was incorrect and invalid? No way! Imagine a bunch

of drunken, 9th century, Scandinavian Vikings being upset if a Lyre (harp) was 8 hZ out of tune....not a chance, they still would have been singing and dancing and having a great time.

What does artistic/creative endeavour mean to you? Is the journey of creation as much a part of the art as the finished 'artifact'?

It gives me purpose and a way to feel like I'm contributing something of value to the world. I'm not a philanthropist or a saint, but I can offer the world art and hopefully make one person's journey a little more special. I do love the process of making it, but I also love experiencing the final product. Feeling the completion and basking in that sense of accomplishment until my brain tells me to get back to work!

In your experience, do you have the sense that your art lives beyond you, that by touching another it continues to breathe and change through them?

If so, what does that mean for you?

Our art is definitely a way to achieve immortality, or at least an extended life. Banksy once said "they say you die twice. Once when you stop breathing and the second, a bit later on, when somebody mentions your name for the last time." I guess you could also say, the second time you die is when someone enjoys your art for the last time. I definitely think about music as my legacy as I don't have any kids yet and I wonder what will be left of me after I depart this world, and hope that people can enjoy my art for many years after I've kicked the proverbial bucket.

Mr.Licka creates sonic landscapes born from seeds he himself has sown, his music is part of him. No separation. No distinction. He has reminded me that our ability to create and to share our creations with the world can show us something of immortality, and that even in death, the echoes of our song will linger on in the timeless lands, forever connected to the source and yet reaching out further into the unknown with courage and beauty.

I'm a Red Team Operator in the Security Assurance team of ABN AMRO bank in The Netherlands. I have an international mindset and a strong affinity with both the technical and strategic aspects of cyber security. It energizes me to place offensive security engagements in a broader organizational context and to share knowledge on a broad range of security-related topics.

Malware Development for Dummies

Todays Jedi arts training by
Cas van Cooten

In the age of EDR, red team operators cannot get away with using pre-compiled payloads anymore. As such, malware development is becoming a vital skill for any operator. Getting started with maldev may seem daunting, but is actually very easy. This workshop will show you all you need to get started!

This repository contains the slides and accompanying exercises for the 'MalDev for Dummies' workshop that will be facilitated at Hack in Paris 2022 (additional conferences TBA). The exercises will remain available here to be completed at your own pace - the learning process should never be rushed! Issues and pull requests to this repo with questions and/or suggestions are welcomed.

 Disclaimer: Malware development is a skill that can -and should- be used for good, to further the field of (offensive) security and keep our defenses sharp. If you ever use this skillset to perform activities that you have no authorization for, you are a bigger dummy than this workshop is intended for and you should skidaddle on out of here.

Workshop Description
With antivirus (AV) and Enterprise Detection and Response (EDR) tooling becoming more mature by the minute, the red team is being forced to stay ahead of the curve. Gone are the times of execute-assembly and dropping unmodified payloads on disk - if you want your engagements to last longer than a week you will have to step up your payload creation and malware development game. Starting out in this field can be daunting however, and finding the right resources is not always easy.

This workshop is aimed at beginners in the space and will guide you through your first steps as a malware developer. It is aimed primarily at offensive practitioners, but defensive practitioners are also very welcome to attend and broaden their skillset.

During the workshop we will go over some theory, after which we will set you up with a lab environment. There will be various exercises that you can complete depending on your current skillset and level of comfort with the subject. However, the aim of the workshop is to learn, and explicitly not to complete all the exercises. You are free to choose your preferred programming language for malware development, but support during the workshop is provided primarily for the C# and Nim programming languages.

During the workshop, we will discuss the key topics required to get started with building your own malware. This includes (but is not limited to):

- The Windows API
- Filetypes and execution methods
- Shellcode execution and injection
- AV and EDR evasion methods

Getting Started

To get started with malware development, you will need a dev machine so that you are not bothered by any defensive tooling that may run on your host machine. I prefer Windows for development, but Linux or MacOS will do just as fine. Install your IDE of choice (I use VS Code for almost everything except C#, for which I use Visual Studio, and then install the toolchains required for your MalDev language of choice:

- C#: Visual Studio will give you the option to include the .NET packages you will need to develop C#. If you want to develop without Visual Studio, you can download the .NET Framework separately.
- Nim lang: Follow the download instructions. Choosenim is a convenient utility that can be used to automate the installation process.
- Golang (thanks to @nodauf for the PR): Follow the download instructions.
- Rust (not supported during workshop): Rustup can be used to install Rust along with the required toolchains.
-

Don't forget to disable Windows Defender or add the appropriate exclusions, so your hard work doesn't get quarantined!

ℹ ***Note:*** *Oftentimes, package managers such as apt or software management tools such as Chocolatey can be used to automate the installation and management of dependencies in a convenient and repeatable way. Be conscious however that versions in package managers are often behind on the real thing! Below is an example Chocolatey command to install the mentioned tooling all at once.*

```
choco install -y nim choosenim go rust vscode visualstudio2019community dotnetfx
```

Compiling programs

Both C# and Nim are compiled languages, meaning that a compiler is used to translate your source code into binary executables of your chosen format. The process of compilation differs per language.

C#

C# code (.cs files) can either be compiled directly (with the csc utility) or via Visual Studio itself. Most source code in this repo (except the solution to bonus exercise 3) can be compiled as follows.

Do you have a technique, a discovery, even a rant yu'd like to share with us and maybe the world?
HVCK wouold love t hear from you. CISO , studentt, researcher, redteamer ,PhD or OCD..
It thast crazy mix that will push this space forward≥,

i Note: *Make sure you run the below command in a "Visual Studio Developer Command Prompt" so it knows where to find csc, it is recommended to use the "x64 Native Tools Command Prompt" for your version of Visual Studio.*

```
csc filename.exe /unsafe
```

You can enable compile-time optimizations with the /optimize flag. You can hide the console window by adding /target:winexe as well, or compile as DLL with /target:library (but make sure your code structure is suitable for this).

Nim

Nim code (.nim files) is compiled with the nim c command. The source code in this repo can be compiled as follows.

```
nim c filename.nim
```

If you want to optimize your build for size and strip debug information (much better for opsec!), you can add the following flags.

```
nim c -d:release -d:strip --opt:size filename.nim
```

Optionally you can hide the console window by adding --app:gui as well.

Golang

Golang code (.go files) is compiled with the go build command. The source code in this repo can be compiled as follows.

```
GOOS=windows go build
```

If you want to optimize your build for size and strip debug information (much better for opsec!), you can add the following flags.

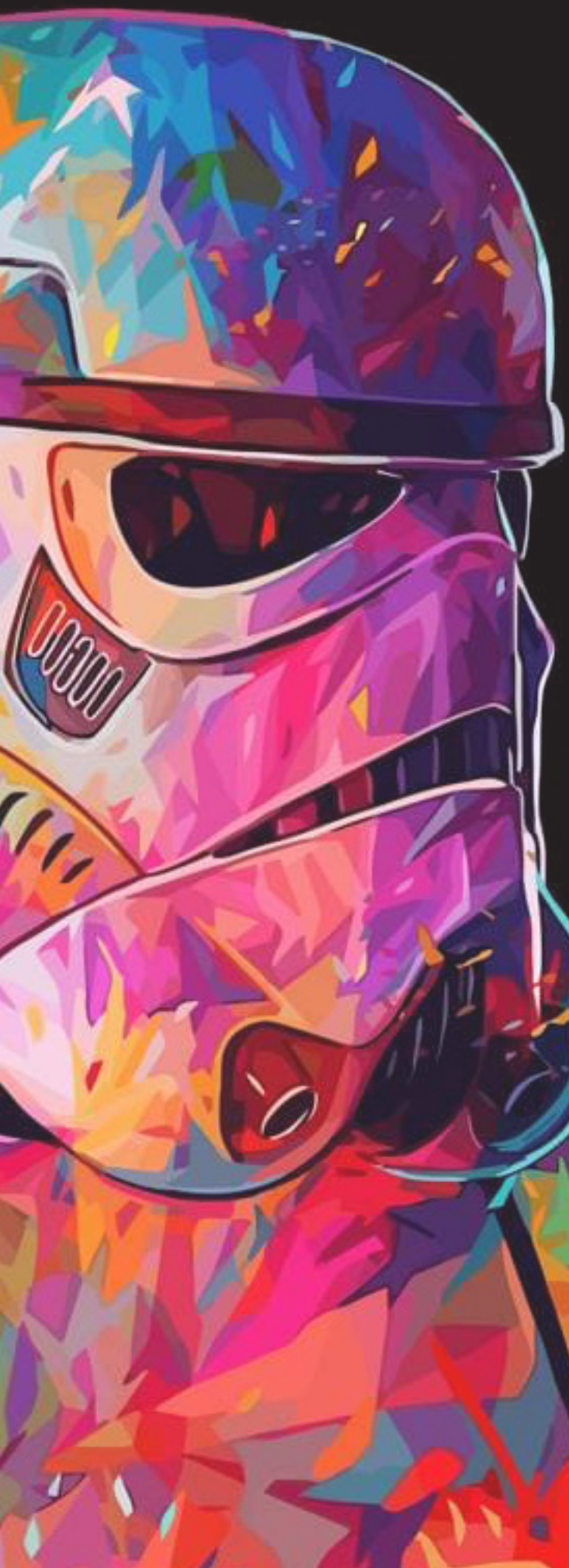
```
GOOS=windows go build -ldflags "-s -w"
```

Dependencies

Nim

Most Nim programs depend on a library called "Winim" to interface with the Windows API. You can install the library with the Nimble package manager as follows (after installing Nim):

```
nimble install winim
```



Golang

Some dependencies are used in the source code of this repo. You can install them as follows (after installing Go):

```
go mod tidy
```

Resources

The workshop slides reference some resources that you can use to get started. Additional resources are listed in the README.md files for every exercise!

https://github.com/chvancooten/maldev-for-dummies
https://casvancooten.com/
https://twitter.com/chvancooten
https://www.linkedin.com/in/chvancooten/

If that blew your skirt up a little, here a few more MalDev links to get you started

.
Exploit and Malware Development:[/b]

1. [CovertUtils](https://github.com/operatorequals/covertutils) && [Documentation](https://covertutils.readthedocs.io/en/latest/)
2. [HideProcess](https://github.com/landhb/HideProcess)
3. [WindowsRegistryRootkit](https://github.com/Cr4sh/WindowsRegistryRootkit)
4. [Shneska003-Rootkit-CerberMasker](https://github.com/MaKiPL/Shneska003-Rootkit-CerberMasker)
5. [booty](https://github.com/ahixon/booty)
6. [TinyXPB](https://github.com/MalwareTech/TinyXPB)
7. [Win64-Rovnix-VBR-Bootkit](https://github.com/m0n0ph1/Win64-Rovnix-VBR-Bootkit)
8. [hidden](https://github.com/JKornev/hidden)
9. [Python-Ransomware](https://github.com/ncorbuk/Python-Ransomware)
10. [PwnTools](https://github.com/ps1337/pwntools-r2) -> [Documentation](https://docs.pwntools.com/en/stable/)
11. [go-shellcode](https://github.com/brimstone/go-shellcode)
12. [Malware-Sample-Library](https://github.com/mstfknn/malware-sample-library)
13. [Awesome-Linux-Rootkits](https://github.com/tkmru/awesome-linux-rootkits)
14. [Malware-and-ExploitDev-Resources](https://github.com/evilbuffer/

Exercise 0

Development Environment

# Description	Set up a development VM with tools for your chosen language.
# Tips	No tips provided for this one □□
# References	Refer to the [‘Getting Started’ section of the main readme](/README.md#getting-started) for guidance.
# Solution	No solution provided for this one □□

Exercise 1 - Basic Shellcode Loader

r## Description

Use `msfvenom` to generate some shellcode, and write a basic loader that executes it in the current process.## TipsThis exercise is throwing you in the deep end by design! If you are lost, start by looking at some open-source examples, then try to replicate what they are doing yourself.> I **Note:** It may be tempting to copy and paste whole examples, but this is not advisable for several reasons:

>

> 1. You learn better by applying the techniques yourself

> 2. Public code is often fingerprinted, doing it your own way will help with evasion

> 3. Some repositories may include malicious code (e.g. bad shellcode) that you may accidentally execute### msfvenomWith `msfvenom`, you can use the `csharp` format for C#, and the `raw` format for Nim. The latter requires you to modify the shellcode to be a Nim byte array (make sure you get the length right):```
var shellcode: array[5, byte] = [byte 0x90, 0x90, 0x90, 0x90, 0x90]
```### Windows API combinationsRemember the various API calls you can use. There are two combinations that make the most sense for this exercise:- [copy memory] + `VirtualProtect()` + `CreateThread()`

This is the most straightforward way to get your shellcode to execute. Because the shellcode is already placed in memory the moment you define a variable, you can "skip" the first step and just target the variable with your shellcode with the `VirtualProtect()` call to make it executable. After that, you can use `CreateThread()` to execute the shellcode (or cast a pointer, see below).- `VirtualAlloc()` + copy memory + `CreateThread()`

This is an alternative to the above, another very popular way of executing shellcode. You can use `VirtualAlloc()` to allocate an executable memory region for the shellcode, and then copy your shellcode into the allocated memory. The result is the same as the first method.Copying memory can be done without API calls using `Marshal.copy` for C# or `copyMem` for Nim.> **Note:** Depending on the type of shellcode you are using, you may need to use the `WaitForSingleObject()` API to keep your program alive while it is running your shellcode. This is only required for long-running shellcodes, such as a CobaltStrike beacon.> If you're feeling adventurous, you can use the native API (Nt-functions from `NTDLL.dll`) counterparts of these functions instead. See also [bonus exercise 1](../BONUS%20Exercise%201%20-%20Basic%20Loader%20Without%20CreateThread/). There are many more API functions to explore as well, for an overview check out malapi.io.### Invoking the Windows API (C# only)C# doesn't have native support for calling the Windows API, so you will have to define the API functions you want to use yourself. This is called P/Invoke. Luckily, most API functions and how to call them have been well documented, e.g. on pinvoke.net.Alternatively, you may opt to dynamically resolve the function calls. While harder to implement, this is much more opsec-safe. The [D/Invoke library](https://github.com/TheWover/DInvoke) can be used to implement this.### Casting pointers - an alternative to `CreateThread()`Instead of using the `CreateThread()` API, you can use a technique called "casting a pointer" to turn your shellcode memory into a function and execute it in the current thread. You can see examples [here (C#)](https://tbhaxor.com/execute-unmanaged-code-via-c-pinvoke/) and [here (Nim)](https://github.com/byt3bl33d3r/OffensiveNim/issues/16#issuecomment-757228116). This avoids calling a suspicious API function, but brings problems of its own (such as the program crashing after your shellcode returns).### Golang tipsThe library `golang.org/x/sys/windows` is the official library of Golang that implements the Windows API. However, some unusual APIs that we are using in malware development may be missing from this library. For example, the `CreateThread` function is not available. To implement this function in our code, we can use the `golang.org/x/sys/windows/mkwinsyscall` package to generate a file (usually [`zsyscall_windows.go`](https://github.com/golang/sys/blob/master/windows/zsyscall_windows.go) generated from [`syscall_windows.go`](https://github.com/golang/sys/blob/c0bba94af5f85fbad9f6dc2e04ed5b8fac9696cf/windows/syscall_windows.go#L168)) that will contain all our Windows APIs implemented in Golang.To generate the right input line for `mkwinsyscall`, we need to get the syntax of the function. Fortunately, this one is documented on [Microsoft](https://docs.microsoft.com/en-us/windows/win32/api/processthreadsapi/nf-processthreadsapi-createthread):

```
HANDLE CreateThread(
  [in, optional]  LPSECURITY_ATTRIBUTES   lpThreadAttributes,
  [in]            SIZE_T                  dwStackSize,
  [in]            LPTHREAD_START_ROUTINE  lpStartAddress,
  [in, optional]  __drv_aliasesMem LPVOID lpParameter,
  [in]            DWORD                   dwCreationFlags,
  [out, optional] LPDWORD                 lpThreadId
);
```

```The corresponding line for `mkwinsyscall` is the following```golang
//sys   CreateThread(lpThreadAttributes *SecurityAttributes, dwStackSize uint32, lpStartAddress uintptr, lpParameter uintptr, dwCreationFlags uint32, lpThreadId *uint32)(threadHandle windows.Handle, err error) = kernel32.CreateThread
```This is basically the prototype of the function with at the end the location of the function in the Windows API, in our case `kernel32.CreateRemoteThread`.

The tricky part is to translate each `C` type into a Golang type. To simplify the process, you can look at the existing lines in Windows package and if something is wrong debug with a tool like [APIMonitor](https://apimonitor.com/) and compare with a working call of the API.Finally, make sure to add the following line in `syscall_windows.go`

```golang
//go:generate go run golang.org/x/sys/windows/mkwinsyscall -output zsyscall_windows.go syscall_windows.go
```And then, the file `zsyscall_windows.go` can be generated with:

```bash
go generate syscall_windows.go
```These steps can be time-consuming, but meanwhile the windows package is updated you can find several of the API already implemented in the [go-windows](https://github.com/nodauf/go-windows) repository.

## References### C#- [Execute Unmanaged Code via C# P/Invoke](https://tbhaxor.com/execute-unmanaged-code-via-c-pinvoke/)

- [Offensive P/Invoke: Leveraging the Win32 API from Managed Code](https://posts.specterops.io/offensive-p-invoke-leveraging-the-win32-api-from-managed-code-7eef4fdef16d)

- [x64ShellcodeLoader.cs](https://gist.github.com/matterpreter/03e2bd3cf8b26d57044f3b494e73bbea)### Nim- [shellcode_loader.nim](https://github.com/sh3d0ww011/nim_shellloader/blob/master/shellcode_loader.nim)

## # Exercise 2 - Basic Shellcode Injector##

DescriptionCreate a new project that injects your shellcode in a remote process, such as `explorer.exe`.## Tips

This exercise is actually very similar to [Exercise 1](../Exercise%201%20-%20Basic%20Shellcode%20Loader/) in terms of implementation. The basic approach is comparable to the `VirtualAlloc()` method we saw there, except this time we are using a different API combination: `OpenProcess()` to get a handle on the target process, `VirtualAllocEx()` to allocate executable memory in the remote process, `WriteProcessMemory()` to copy the shellcode into the allocated, and `CreateRemoteThread()` to execute the shellcode as part of the target process.> If you're feeling adventurous, you can use the native API (Nt-functions from `NTDLL.dll`) counterparts of these functions instead. Alternatively, look at other ways to expose your shellcode to the target process' memory, such as `NtCreateSection()` and `NtMapViewOfSection()` (example [here](https://www.ired.team/offensive-security/code-injection-process-injection/ntcreatesection-+-ntmapviewofsection-code-injection)).### Getting a handleKeep in mind that in order to get a handle, we need to have sufficient privileges over the target process. This generally means that you can only get a handle for a process owned by the current user, and not those owned by other users or managed by the system itself (makes sense right?). However, if you are executing from a privileged context (i.e. running as `SYSTEM` or with the `SeDebugPrivilege` enabled) you can get a handle to any process, including system processes. When designing malware that injects remotely, you need to be conscious about the target process that you choose. Choosing the wrong process may cause your malware to fail because the process is not present, or you have insufficient privileges. Furthermore, injecting from a privileged context into a low-privileged process will drop your privileges.> I **Note:** This is why making the target process configurable and basing it on the target environment is a good idea. You may hardcode the name or process ID of `explorer.exe` for now, we will improve that functionality in [bonus exercise 2](../BONUS%20Exercise%202%20-%20Basic%20Injector%20With%20Dynamic%20Target/).### Golang tipsThe library `golang.org/x/sys/windows` is the official library of Golang that implements the Windows API. However, some unusual APIs that we are using in malware development may be missing from this library. For example, the `VirtualAllocEx` or `CreateRemoteThread` functions are not available.To implement these functions in our code, we can use the `golang.org/x/sys/windows/mkwinsyscall` package to generate a file (usually [`zsyscall_windows.go`](https://github.com/golang/sys/blob/master/windows/zsyscall_windows.go) generated from [`syscall_windows.go`](https://github.com/golang/sys/blob/c0bba94af5f85fbad9f6dc2e04ed5b8fac9696cf/windows/syscall_windows.go#L168)) that will contain all our Windows APIs implemented in Golang.To generate the right input line for `mkwinsyscall`, we need to get the syntax of the function. Fortunately, this one is documented on [Microsoft](https://docs.microsoft.com/en-us/windows/win32/api/processthreadsapi/nf-processthreadsapi-createremotethreadex):

```
HANDLE CreateRemoteThread(
  [in]  HANDLE                 hProcess,
  [in]  LPSECURITY_ATTRIBUTES  lpThreadAttributes,
  [in]  SIZE_T                 dwStackSize,
  [in]  LPTHREAD_START_ROUTINE lpStartAddress,
  [in]  LPVOID                 lpParameter,
  [in]  DWORD                  dwCreationFlags,
  [out] LPDWORD                lpThreadId
);
```

```The corresponding line for `mkwinsyscall` is the following```golang
//sys   CreateRemoteThread(process Handle, threadAttributes *SecurityAttributes, stackSize uintptr, startAddress uintptr, lpParameter uintptr, dwCreationFlags uint32, lpThreadId *uint32) (threadHandle windows.Handle, err error) = kernel32.CreateRemoteThread
```This is basically the prototype of the function with at the end the location of the function in the Windows API, in our case `kernel32.CreateRemoteThread`.

The tricky part is to translate each `C` type into a Golang type. To simplify the process, you can look at the existing lines in Windows package and if something is wrong debug with a tool like [APIMonitor](https://apimonitor.com/) and compare with a working call of the API.Finally, make sure to add the following line in `syscall_windows.go`

```golang
//go:generate go run golang.org/x/sys/windows/mkwinsyscall -output zsyscall_windows.go syscall_windows.go
```And then, the file `zsyscall_windows.go` can be generated with:

```bash
go generate syscall_windows.go
```These steps can be time-consuming, but meanwhile the windows package is updated you can find several of the API already implemented in the [go-windows](https://github.com/nodauf/go-windows) repository

## References### C#- [A simple Windows code injection example written in C#](https://andreafortuna.org/2019/03/06/a-simple-windows-code-injection-example-written-in-c/)### Nim- [shellcode_bin.nim](https://github.com/byt3bl33d3r/OffensiveNim/blob/master/src/shellcode_bin.nim)## SolutionExample solutions are provided in the [solutions folder](solutions/). ([C#](solutions/csharp/), [Nim](solutions/nim/)). Keep in mind that there is no "right" answer, if you made it work that's a valid solution!

THESE ARE THE EXERCISES RIGHT HERE IF YOU'RE SUPER KEEN TO GET YOU'RE STUXNET ON..

NO ITS NOT A PISS TAKE. THE TEXT IS SAVED AS A VECTOR IMAGE. ZOOM IN.. SEE FOR YOURSELF.
```

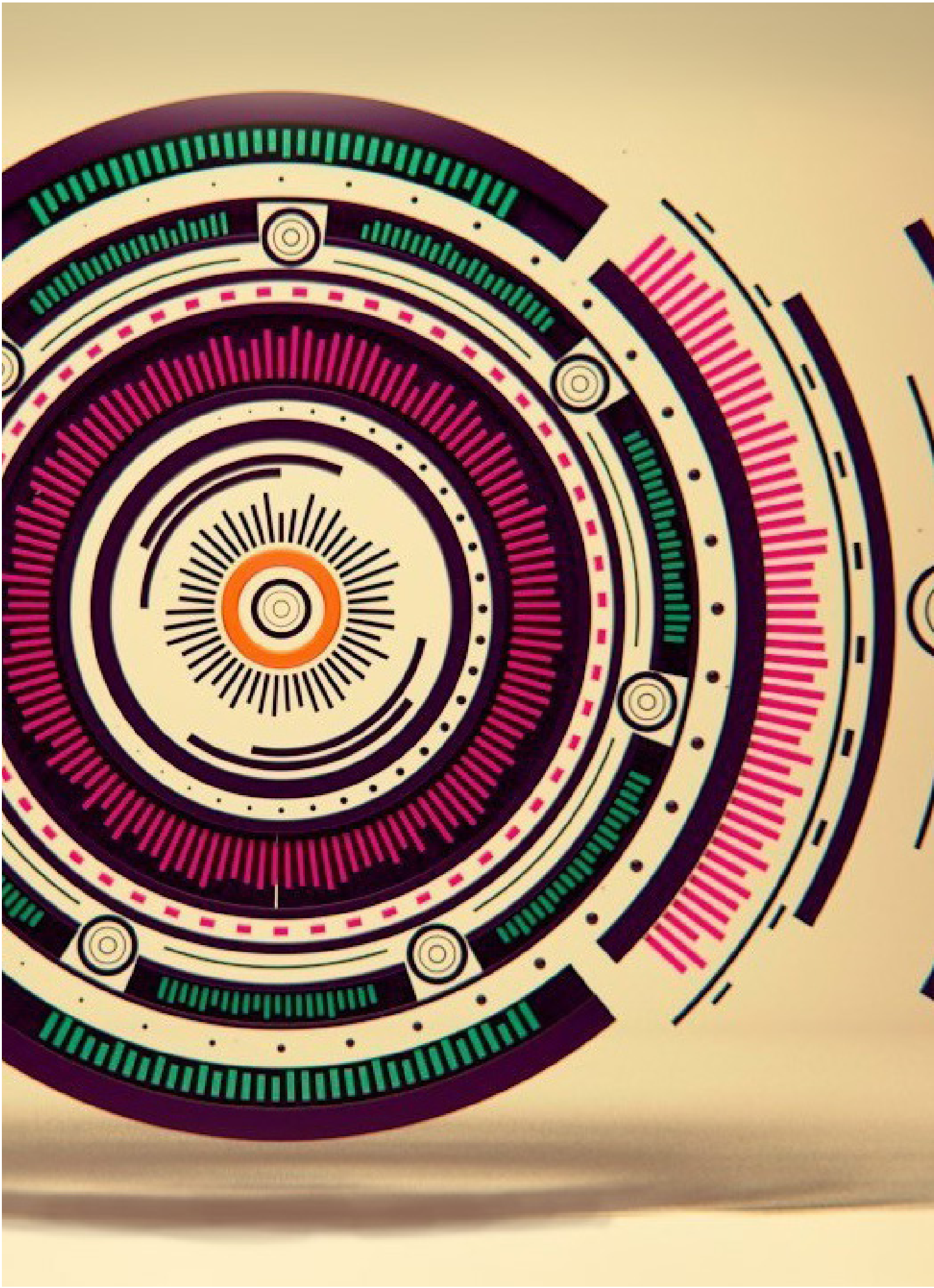
curated by

Lil Red

HV

Arts


P8R
H8R

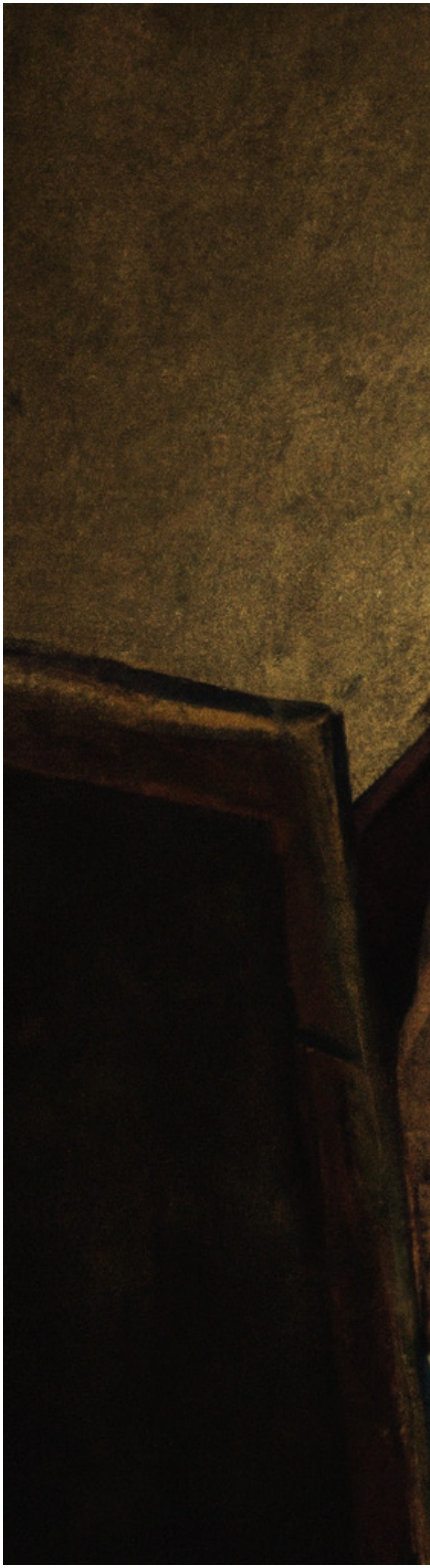


‘BEING HONOURED WITH BETA ACCESS TO **OPENAI’S DALLE2**, I STARTED TO EXPLORE TYPICAL CYBER SECURITY CONCEPTS. THE SYSTEM IS A TEXT-TO-IMAGE GENERATOR WITH A LOT OF POTENTIAL AND POWER. YOU ENTER A TEXT, E.G. ‘THE MONA LISA AS A HACKER’, AND A FEW SECONDS LATER YOU ARE PRESENTED WITH 6 IMAGES THAT THE AI CREATED FROM YOUR TEXT. THAT’S HOW THE IMAGES BELOW WERE CREATED. ALL RIGHTS ARE WITH OPENAI, THIS IS JUST A SHOWCASE OF WHAT THEIR SYSTEM IS CAPABLE OF DOING, USING A FEW OF MY FAVORITE IMAGES..

MAXIMILIAN HEINEMEYER

CHIEF PRODUCT OFFICERE - DARKTRACE

‘THE MONA LISA AS A HACKER’



‘A CYBER INCIDENT, KNITTED’

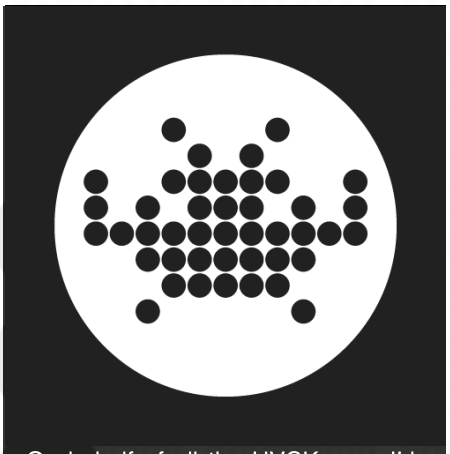

NERT
RESHER

‘CHARMING KITTEN CYBER ACTOR, DIGITAL ART’

‘AN AI DEFENDING A NETWOR

IN CYBERSPACE’

‘**DALLE2’S** GENERAL CONCEPT OF ‘CYBER’ IS OFTEN STYLIZED NUMBERS IN NEON COLOURS, OFTEN ON COMPUTER SCREENS OR IN FRONT OF A BLACK SCREEN. A **‘HACKER’** IS USUALLY A MAN WITH A BLACK HOODIE IN FRONT OF A SCREEN. WE KNOW THESE TROPES FROM MAIN-STREAM MEDIA REPORTING ON THE CYBER SECURITY INDUSTRY - SEEING **DALLE2** USE FAMILIAR IMAGERY COMES THEREFORE AS NO SURPRISE. IT IS STILL INTERESTING TO SEE THE MORE ABSTRACT CONCEPTS LIKE **A KNITTED CYBER INCIDENT** , THE **MONA LISA BEING A HACKER** OR YOUR FAVORITE **APT GROUP AS DEPICTED BY AN AI.** THANKS TO **OPENAI** FOR CREATING THIS TECHNOLOGY AND GRANTING ME EARLY ACCESS TO IT.’



On behalf of all the HVCK crew I’d just like to thank Max for going out of his way to organise those images from Dalle2. In a fashion befiting a cyber juggernaught such as himself he went above and beyond the call of duty and displayed unquenchable passion for technology.

Fo these reasons we award you the “Triple Chuck”. The highest accolade one can achieve as a biological human. We considered the “quad chuck” but we are saving that for the first person to tweet with Neuralink.

Congratulations mate :)

- d8rh8r

FOR THOSE CURIOUS AND WANT TO DIG DEEPER

HTTPS://ARXIV.ORG/ABS/2204.06125

HTTPS://OPENAI.COM/ABOUT/

is alert fatigue a problem in your SOC

maybe there's a better way to do things?

SMARTCYBER solutions

Lil Red

Lil Red

Stay frosty out there friends

Lots of locce from
d8rh8r & the hvck team

big thank yous to
Fiona Lewis
Aaron the sdr mna
Mr Sukshine
David Lee
Rio the Animus
Gareth STP
Max
Dalle2
Cas van Coomten
Mr Licka

a special thank you to the quark
you belief in me and your endless patience can never be repaid... Thank you